VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Uzamykací obrazovka pro Android používající biometriky

Biometric-based Android Lock Screen

2014 Bc. Rostislav Vítek

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Zadání diplomové práce

Student: **Bc. Rostislav Vítek**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2612T025 Informatika a výpočetní technika

Téma: Uzamykací obrazovka pro Android používající biometriky
Biometric-based Android Lock Screen

Zásady pro vypracování:

Cílem této práce je seznámit se s typickými metodami fyziologických a behaviorálních biometrik a vybrat alespoň 2 (dostatečně rychlé) metody, které budou použity pro ověření identity uživatele a odemčení telefonu.

1. Nastudujte běžné i méně známé metody biometrické verifikace.
2. Zjistěte, jaké metody biometrické verifikace jsou již v prostředí OS Android implementovány.
3. Vyberte dvě metody biometrické verifikace a implementujte je v podobě knihovny, umožňující přidávat další metody prostřednictvím definovaného API. Vytvořte odemykací obrazovku pro Android, která bude knihovnu využívat.
4. Otestujte spolehlivost a rychlost implementovaných metod, srovnejte dosažené výsledky s integrovanou metodou rozpoznání obličeje a vyhodnoťte případná bezpečnostní rizika.

Seznam doporučené odborné literatury:

[1] A. K. Jain, P. Flynn, A. A. Ross. Handbook of Biometrics. Springer, 2008, ISBN 978-0-387-71040-2.
[2] J. R. Kwapisz. Cell phone-based biometric identification. Fourth IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS), 2010, DOI: 10.1109/BTAS.2010.5634532.
[3] V. Kanhangad, A. Kumar, A, D. Zhang. Contactless and pose invariant biometric identification using hand surface. IEEE Transactions on Image Processing, Vol. 20(5), DOI: 10.1109/TIP.2010.2090888.
[4] Meier, R. Professional Android 2 Application Development. Wrox Press Ltd., 2010. ISBN: 978-0-47056-552-0.
[5] John C. Russ: The Image Processing Handbook, Sixth Edition, CRC Press, 2011, ISBN 1439840636.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Pavel Moravec, Ph.D.**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014

doc. Dr. Ing. Eduard Sojka
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

„Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“

V Ostravě dne 6. května 2014

Rostislav Vítek

Poděkování

Rád bych poděkoval vedoucímu diplomové práce Ing. Pavlu Moravcovi, PhD. za odbornou pomoc a konzultaci při vytváření této práce.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Poděkování

Tato práce byla vypracována s podporou projektu Bio-inspirované metody: věda, vzdělávání a transfer znalostí, reg. č. CZ.1.07/2.3.00/20.0073 podpořeného Operačním programem Vzdělávání pro konkurenceschopnost, financovaného ze strukturálních fondů EU a státního rozpočtu ČR.

Abstrakt

Diplomová práce v úvodní části vysvětluje základní pojmy z oblasti biometrie, popisuje fungování a praktickou aplikaci biometrických systémů a podává přehled o známých i méně rozšířených biometrických metodách. V rámci práce je představen současný stav používání biometrických metod v prostředí OS Android. V praktické části aplikace je detailně popsán průběh vývoje aplikace pro zařízení fungující na platformě Android. Aplikace realizuje chování uzamykací obrazovky a poskytuje zvýšenou úroveň zabezpečení zařízení využitím biometrických metod ověřování identity. Implementované biometrické metody jsou součástí samostatné biometrické knihovny, jejíž vývoj je v práci rovněž popsán. V implementovaných metodách jsou zastoupeny dva způsoby ověřování identity, a to statický i dynamický. Součástí práce je závěrečné otestování a ohodnocení spolehlivosti implementovaných ověřovacích metod.

Klíčová slova

Biometrie, biometrická charakteristika, biometrický systém, identita, verifikace, identifikace, Android, knihovna OpenCV, uzamykací obrazovka.

Abstract

The thesis explains basic biometric terms, describes function and application of biometric systems and presents an overview of well known biometric methods as well as less known methods. The state of the art biometric methods used in Android environment are presented within the thesis. In the practical part of the thesis is described an implementation of the application developed for Android devices. The application implements behavior of a lock screen and provides higher level of security by using biometric verification methods. Implemeted biometric methods are components of standalone biometric library and represent both static and dynamic ways of identity verification. Developement of the library is described within the thesis as well. The methods are tested and their reliability is evaluated in the end of the thesis.

Key words

Biometrics, biometric characteristic, biometric system, identity, verification, identification, Android, OpenCV library, lock-screen.

Seznam použitých symbolů a zkratek

API	Application Programming Interface
APK	Android package
BSD	Berkeley Software Distribution
DET	Detection Error Tradeoff
DVM	Dalvik Virtual Machine
ERR	Equal Error Rate
FAR	False Accept Rate
FIDO	Fast Identity Online
FMR	False Match Rate
FNMR	False Non Match Rate
FRR	False Reject Rate
FTA	Failure To Acquire
FTE	Failure To Enroll
FTM	Failure To Match
LIFO	Last in first out
MB	Megabyte
Mpix	Megapixel
OHA	Open Handset Alliance
OS	Operační systém
PC	Personal computer (osobní počítač)
ROC	Receiver Operating Characteristic

Seznam tabulek

Seznam obrázků

Obsah

1 Úvod

V současné době se i několikrát za den setkáváme s nutností prokázat svou totožnost. Jinými slovy musíme často dokazovat, že jsme skutečně tím, za koho se vydáváme. V legislativní oblasti nám k tomu slouží nejrůznější doklady, jako jsou občanské průkazy, řidičské průkazy nebo cestovní pasy. V zaměstnání nebo ve škole používáme přístupové karty a klíče, v soukromí jsme pak nuceni si pamatovat množství uživatelských jmen, hesel a PIN kódů, abychom mohli přistupovat k online účtům, používat internetové a telefonní bankovnictví nebo třeba jen zapnout svůj mobilní telefon či notebook. Pamatovat si přístupové údaje ke všem svým účtům se stává neúnosným. Navíc pokud se někomu podaří získat naše tajné heslo nebo zfalšovat naše identifikační doklady, získá tím přístup k našim citlivým datům a může před autoritami vystupovat pod naším jménem, což nám může způsobit řadu problémů a nepříjemností.

Biometrie je věda, která se zabývá měřením tělesných (fyziologických, behaviorálních a chemických) vlastností jedinců a umožňuje je pomocí těchto vlastností jednoznačně identifikovat. V současné době již existují tzv. biometrické systémy, které dokáží automatizovaně získávat vybrané charakteristiky a rozhodovat, zda je uživatel skutečně tím, kým tvrdí. Biometrie přináší zvýšení bezpečnosti a zároveň komfortu uživatelů.

Biometrickou charakteristikou může být jakákoliv měřitelná lidská vlastnost, která je dostatečně univerzální (má ji každý jedinec z cílové populace), můžeme na jejím základě spolehlivě rozlišit většinu jedinců, je dostatečně neměnná v čase, lze ji v rozumném čase změřit a zpracovat, je odolná vůči pod-vrhu a uživatelé jsou ochotni tuto vlastnost systému k měření poskytnout. V současnosti se k ověřování identity používá již mnoho biometrických charakteristik, jmenujme například otisk prstu, geometrii ruky nebo ucha, hlas, obličej, duhovku, DNA, podpis nebo styl chůze.

Diplomová práce popisuje vývoj mobilní aplikace pro zařízení s operačním systémem Android. Aplikace simuluje činnost zámku obrazovky a k odemknutí zařízení využívá biometrických metod ověřování identity. Metody jsou implementovány v rámci samostatné knihovny, jejíž vývoj je v diplomové práci rovněž detailně popsán.

Kapitola, která následuje po úvodu, seznamuje čtenáře s biometrií, vysvětluje základní pojmy, fungování biometrických systémů, detailněji popisuje vybrané biometrické charakteristiky a hodnotí vhodnost metod k implementaci v rámci práce. Následující kapitola se zabývá popisem využívaných technologií, tedy platformy Android a grafické knihovny OpenCV. V rámci kapitoly je vylíčen současný stav užívání biometrik v prostředí Android. Čtvrtá kapitola dopodrobna analyzuje biometrické charakteristiky zvolené k implementaci. Pátá kapitola představuje náhledy uživatelského rozhraní vyvíjené aplikace a navrhuje vhodné rozdělení projektu na komponenty. Kapitola číslo šest se věnuje konkrétním postupům, které byly užity při vývoji biometrické knihovny i zámku obrazovky. Obsahem poslední kapitoly je pak popis průběhu testování implementovaných metod ověřování identity a srovnání výsledků s metodou ověřování identity pomocí obličeje, která je již v prostředí Android realizována.

2 Seznámení s biometrií

Tato kapitola vysvětluje pojem *biometriky,* popisuje různé druhy biometrických údajů a u každého z nich je zhodnocena vhodnost použití pro problém řešený v diplomové práci. Dále je popsáno fungování biometrického systému a objasněn rozdíl mezi verifikací a identifikací.

2.1 Biometriky a biometrické systémy

Podle [1] představuje pojem biometriky soubor automatizovaných metod pro ověření identity jedince za pomocí měření jeho fyziologických, chemických nebo behaviorálních znaků či charakteristik. Souhrnně jsou tyto charakteristiky označovány jako biometrické charakteristiky, znaky či údaje. Mezi fyziologické biometrické znaky se řadí například papilární linie na posledních článcích prstů, hlas[1], obličej, duhovka, sítnice nebo tvar lidského ucha či dlaně. Metody ověřování těchto charakteristik můžeme označit jako statické. Jako behaviorální charakteristiky jsou označovány rysy v chování – např. styl chůze, frekvence úderů při psaní na klávesnici nebo podpis, které jsou ověřovány dynamickými metodami. Pach nebo DNA zastupují chemické znaky. Vybrané biometrické charakteristiky jsou blíže popsány v podkapitole 2.2.

Jak uvádí [2, s. 3], biometrický systém je systém, který od člověka získává biometrické charakteristiky, z nich extrahuje sadu klíčových znaků[2], kterou porovnává se sadou uloženou v databázi, a nakonec provádí naprogramované funkce podle výsledku porovnávání. Biometrický systém můžeme tedy rozdělit na čtyři hlavní komponenty, a to *senzorový modul, modul stanovení kvality a extrakce klíčových znaků, srovnávací modul* a *databázový modul.* V chování systému lze rozlišit dvě základní funkcionality – registrační a ověřovací.

1. **Senzorový modul** – jedná se o tu část systému, která je odpovědná za získávání biometrických dat od uživatele. Typicky je reprezentován kamerou, fotoaparátem, mikrofonem apod.

2. **Modul stanovení kvality a extrakce klíčových znaků** – tento modul se stará v první řadě o posouzení vhodnosti biometrických dat k dalšímu zpracování. To znamená, že modul určuje, zda data získaná senzorovým modulem mají dostatečnou kvalitu pro korektní práci v dalších fázích. Pokud je kvalita dat příliš nízká, je potřeba, aby uživatel poskytl data znovu. Pokud je kvalita dostatečná, jsou extrahovány klíčové charakteristiky (biometrický markant)

[1] Hlas řadíme mezi fyziologické i behaviorální charakteristiky, je totiž dán nejen tvarem a velikostí ústrojí, která produkují hlas, ale také artikulací jedince. [4, s. 33]
[2] Podle [3, s. 15] nazýváme extrahované charakteristiky pojmem *biometrický markant.*

poskytnutých biometrických dat. V případě, že systém pracuje v registračním módu, je biometrický markant uložen jako vzor do databáze.

3. **Srovnávací modul** – biometrický markant je porovnáván se šablonou, která je uložena v databázi, a je počítáno skóre určující míru podobnosti. Součástí odpovědností srovnávacího modulu je na základě skóre rozhodnout, zda označit identitu uživatele za ověřenou.

4. **Databázový modul** – uchovává biometrické markanty extrahované z biometrických údajů spolu s dalšími údaji (např. jméno, adresa, identifikační číslo…), které charakterizují uživatele.

Oproti klasickým metodám ověřování identity, jako jsou například cestovní pasy nebo kombinace uživatelského jména a hesla, poskytují biometriky zvýšený stupeň zabezpečení, neboť biometrické údaje nelze jednoduchým způsobem falšovat a je prakticky nemožné je odcizit. Navíc jsou často přívětivější k uživatelům, kteří nejsou nuceni pamatovat si hesla nebo nosit při sobě identifikační doklady.

2.1.1 Identita, verifikace versus identifikace

Základním pojmem, který je potřeba v úvodu práce definovat, je pojem *identita*. Jak uvádí [3, s. 10], *„identita je jednoznačná charakteristika každého z nás. Je potřeba však rozlišovat fyzickou a elektronickou identitu.“* Fyzickou identitu, která je dána našimi fyziologickými a behaviorálními znaky, máme pouze jednu. Elektronických identit si můžeme vytvořit libovolné množství. Takové identity jsou dány například kombinací jedinečného uživatelského jména a příslušného hesla na různých internetových serverech.

Podle [4, s. 10] může biometrický systém fungovat ve dvou módech ověřování identity uživatele, a to buď v módu verifikačním, nebo identifikačním. Pojmy verifikace a identifikace je potřeba vysvětlit a rozlišovat pro správné pochopení dalších částí diplomové práce.

Proces verifikace funguje tak, že se uživatel systému prezentuje za pomoci své elektronické identity, a pokud je daná osoba v systému registrovaná, systém následně vyhledá ve své databázi odpovídající biometrické údaje. Tato data jsou porovnána s daty, které poskytl uživatel a v případě kladného výsledku je identita úspěšně ověřena – uživatel je verifikován. Základním rysem procesu verifikace je tedy porovnávání 1:1. [3, s. 10]

Identifikace (lze se setkat i s pojmem autentizace) je oproti tomu proces, kdy uživatel svou identitu nesděluje a poskytne systému pouze své biometrické údaje. Systém poté porovnává tato data se všemi záznamy ve své databázi, dokud nenalezne shodný vzorek. Pokud se to nepodaří, identita uživatele nemůže být ověřena, v opačném případě je uživatel úspěšně identifikován. Principem identifikace je tedy porovnávání 1:N. [3, s. 10]

Aplikace vyvíjená v práci bude fungovat jako zámek obrazovky Android zařízení, které je většinou pevně svázáno s jedním uživatelem. Aplikace bude tedy fungovat ve verifikačním módu, ale nebude po uživateli vyžadovat proklamování jeho identity před poskytnutím biometrické charakteristiky.

2.1.2 Praktická aplikace

Biometrické systémy našly své využití v praxi v mnoha aplikacích, které se dají podle [5] rozdělit do tří oblastí – forenzní, vládní a komerční. Každá oblast má svá specifika a jiné nároky na fungování biometrického systému. Následuje bližší popis využívání biometrických systémů v jednotlivých oblastech:

1. **Forenzní**[3] – v této oblasti se biometrické metody používají již přes 100 let. Pomáhají při vyšetřování kriminálních činů, identifikaci mrtvých nebo při hledání pohřešovaných osob. Biometrické systémy přináší vyšetřovatelům velkou časovou úsporu oproti ruční práci, neboť dokáží mnohem rychleji zpracovat a porovnat veliké množství záznamů. Navíc jsou biometrické systémy schopny dosáhnout větší přesnosti než pouhé lidské oko, které při porovnávání některých biometrických údajů nemůže rozpoznat rozdíly do takových detailů.

2. **Vládní** – celosvětový bezpečnostní problém spočívá ve velkém množství padělaných nebo odcizených identifikačních dokladů. V současnosti již mnohá mezinárodní letiště[4] používají systémy pro identifikaci osob pomocí oční duhovky nebo například otisku prstu, aby bylo pasažérům zamezeno cestovat do jiného státu pod falešnými údaji. Pro zvýšení bezpečnosti jsou i do stávajících dokladů přidávány biometrické údaje[5], které nelze tak jednoduše ukrást nebo padělat.

3. **Komerční** – stále větší množství citlivých dat (např. lékařské záznamy, informace o kreditních kartách, osobní data v osobních počítačích nebo mobilních zařízeních) je v současnosti ukládáno v elektronické podobě, čímž se zvýšily také nároky na zabezpečení těchto dat. Biometrické systémy dokáží zvýšit úroveň bezpečnosti oproti tradiční kombinaci uživatelských jmen a hesel. Pokrok ve vývoji biometrických systémů umožnil jejich nízkonákladovou instalaci do bankomatů, PC, notebooků, chytrých mobilních telefonů atd.

Aplikace implementovaná v rámci diplomové práce řeší ochranu dat v mobilním telefonu pomocí biometrického zámku obrazovky, spadá tedy do kategorie komerčních biometrických systémů.

---

[3] Dle [35] „vztahující se k použití vědeckých a technologických postupů při zjišťování, prokazování a vyšetřování skutečností a ověřování důkazů v rámci trestního i občanského práva“.
[4] Podle [39] již 28 % světových letišť používá biometriky jako součást svých bezpečnostních systémů.
[5] Podle [34] musí od 1. září 2006 všechny cestovní doklady vydané v Evropské unii s platností delší než 12 měsíců obsahovat strojově čitelné a biometrické údaje. Dle [38] používá cestovní pasy s biometrickými údaji 93 zemí ke dni 28. února 2012.

2.1.3 Chyby a hodnocení biometrických systémů

Tato podkapitola popisuje, jakým způsobem lze hodnotit spolehlivost biometrického systému a jaké typy chyb v biometrických systémech rozlišujeme.

Při klasickém ověřování identity heslem se hledá přesná shoda dvou alfanumerických řetězců. Naproti tomu v biometrických systémech jen zřídka nastane situace, kdy by dva biometrické markanty byly naprosto totožné.[6] Je to logické, neboť je v podstatě nemožné zajistit absolutně stejné podmínky při každém snímání biometrické charakteristiky. Faktory, které snímání ovlivňují, mohou být buď vnější (osvětlení, hluk, nečistoty na senzorovém modulu apod.) nebo uživatelské (zranění, nemoc, odlišný způsob interakce se senzorem, stárnutí atd.). Rozdíly, které můžeme nalézt mezi jednotlivými markanty jednoho a téhož uživatele, nazýváme pojmem *vnitrotřídní variabilita.* Tato míra by měla být u biometrické charakteristiky co nejnižší, ideálně nulová. Naopak rozdíly mezi různými uživateli vyjadřuje pojem *mezitřídní variabilita*, která by měla být pro vybranou biometrickou charakteristiku co možná největší. [2, s. 6-7]

Podle [3, s. 84] je výsledkem porovnávání šablony a nového markantu míra shody nebo taky tzv. *skóre porovnání*. Toto skóre značíme *s.* Jedná se o metriku[7], která udává podobnost šablony s aktuálním vzorkem. Poté je potřeba, aby byl nastaven *práh* T, který rozdělí metriku na dva intervaly. Biometrický systém rozhoduje, zda označit uživatelovu identitu za ověřenou, na základě skóre porovnání s a prahu T:

- Jestliže $s < T$, je výsledkem zamítnutí uživatele,
- jestliže $s \geq T$, je výsledkem přijetí uživatele, jeho identita je ověřena.

Dle [6, s. 6] vykazuje biometrický systém dva základní typy chyb:

- **Chybné přijetí** – nastává v případě, že jsou dva biometrické markanty dvou rozdílných osob považovány za markanty stejné osoby.
- **Chybné odmítnutí** – nastává v případě, že dva biometrické markanty stejné osoby jsou považovány za markanty dvou různých osob.

Na základě četnosti výskytu těchto chyb pak můžeme hodnotit spolehlivost biometrického systému. Odvozujeme dvě základní míry – míru chybného přijetí a míru chybného odmítnutí.

Míra chybného přijetí (*False Accept Rate*, dále jen **FAR**) udává, kolik vzorků vzhledem k celkovému počtu porovnání rozdílných vzorků bylo systémem nesprávně označeno jako shodné

[6] V případě, že systém narazí na absolutní shodu mezi vzorem a nově získaným biometrickým markantem, se někomu s vysokou pravděpodobností podařilo odcizit šablonu a snaží se ji použít pro neoprávněné vniknutí do systému.

[7] Metriky mohou být voleny libovolně, často jde o procentuální vyjádření, kdy 100 % značí absolutní shodu a 0 % dva naprosto rozdílné markanty.

se šablonou, přestože šablona a daný vzorek nepocházely od stejné osoby. Jinými slovy vyjadřuje FAR pravděpodobnost, s jakou se útočníkovi podaří vniknout do systému deklarováním falešné identity.

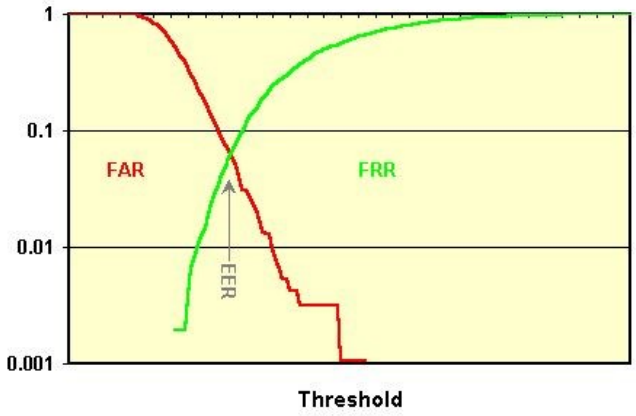
FAR = počet nesprávně ověřených identit uživatelů / celkový počet ověřovacích procesů. [3, s. 86]

Míra chybného odmítnutí (*False Reject Rate*, dále jen **FRR**) je vyjádření pravděpodobnosti, s jakou biometrický systém vyhodnotí markant uživatele jako neshodný se vzorem, přestože pochází oba vzorky od stejného uživatele.

FRR = počet chybných porovnání vzorků stejné osoby / celkový počet porovnání vzorků jedné osoby. [3, s. 86]

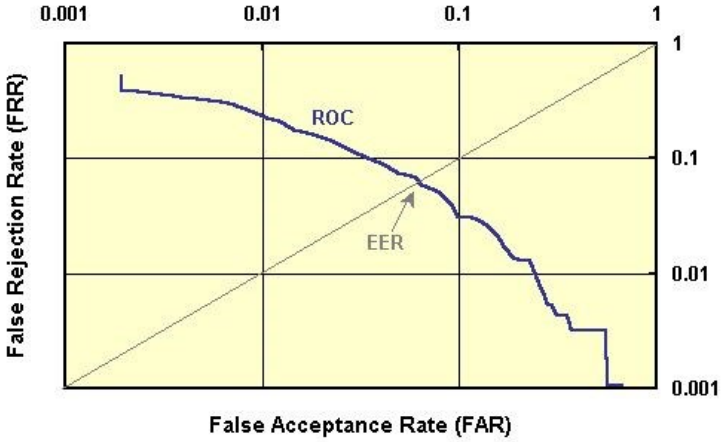
Mimo výše popsané existují i další míry určující kvalitu biometrického systému. **Míra neschopnosti zaregistrovat** (*Failure To Enroll*, dále jen **FTE**) značí podíl uživatelů, jejichž biometrické údaje se nepodařilo z nějakého důvodu uložit. Často tato situace nastává u systémů s kontrolou kvality, FTE tedy může indikovat, do jaké míry je systém schopen pracovat s nekvalitními vzorky. **Míra neschopnosti nasnímat** (*Failure To Acquire*, dále jen **FTA**) představuje četnost výskytu situace, kdy senzor nedokáže přečíst biometrickou charakteristiku. FTA tak slouží jako indikátor kvality senzorového modulu. **Míra neschopnosti porovnat** (*Failure To Match*, dále jen **FTM**) udává, jak často systém není schopen učinit rozhodnutí při porovnávání vzorku se šablonou. **Míra chybné shody** (*False Match Rate*, dále jen **FMR**) a **míra chybné neshody** (*False Non Match Rate*, dále jen **FNMR**) jsou velice podobné mírám FAR, resp. FRR, s tím rozdílem, že neberou v potaz neúspěšné výsledky porovnání z důvodů neschopnosti zaregistrovat nebo nasnímat. [3, s. 87-89]

Míry FMR a FNMR (stejně tak FAR a FRR) jsou obě ovlivňovány prahem T. Čím výše je práh stanoven, tím „přísnější“ je systém v porovnávání. Zvyšuje se FNMR (FRR) a naopak snižuje FMR (FAR), viz Obrázek 2-1. Ne všechny aplikace biometrických systémů mají stejné požadavky na výše chybových měr. Forenzní aplikace většinou vyžadují co možná nejnižší FNMR, např. při identifikaci podezřelých je žádoucí, aby byl označen širší okruh osob tak, aby žádný eventuální pachatel nebyl vynechán. Oproti tomu pro aplikace z oblasti bezpečnostních systémů je stěžejní mírou FMR. Zde je přijatelnější odmítnout uživatele s povolením přístupu (a provést identifikační proces jiným způsobem) než chybně povolit přístup cizím osobám. [6, s. 8]



Obrázek 2-1: Závislost FAR a FRR na prahu T, převzato z [7]

Výkon biometrického systému je popisován pomocí tzv. **ROC** (*Receiver Operating Curve*) křivky, která je vykreslením závislosti FRR a FAR, případně závislosti FMR a FNMR. Ekvivalentem ROC křivky je křivka **DET** (*Detection Error Tradeoff*), která se od ROC odlišuje pouze ve způsobu zanášení dat do grafu. [4, s. 10] Obrázek 2-2 ukazuje příklad ROC křivky.



Obrázek 2-2: ROC křivka, převzato z [7]

Důležitým pojmem je také **míra vyrovnání chyb** (*Equal Error Rate*, dále jen EER). Podle [2, s. 10] vyjadřuje tato míra bod na křivce DET, kdy jsou si FAR a FRR rovny (viz Obrázek 2-1 a Obrázek 2-2). Čím nižší je hodnota ERR, tím výkonnější je biometrický systém.

Všechny míry popsané v této kapitole mohou sloužit jako prostředek pro ohodnocení kvality vybraného biometrického systému. Záleží na konkrétní aplikaci, která míra bude mít v daný moment největší váhu. Při rozhodování o nasazení systému do praxe slouží tyto míry jako jedno z hlavních rozhodovacích kritérií[8].

2.2 Biometrické charakteristiky

V této podkapitole jsou blíže popsány vybrané biometrické charakteristiky a posouzena jejich vhodnost pro implementaci v diplomové práci. Podle [6, s. 4] může být jako biometrická charakteristika sloužit jakákoliv fyziologická nebo behaviorální vlastnost, která splňuje následující podmínky:

- *Univerzálnost* – každý člověk má tuto vlastnost.
- *Rozlišitelnost* – jakékoliv dvě osoby můžeme rozlišit na základě dané vlastnosti.
- *Trvalost* – daná vlastnost je dostatečně trvalá v čase.
- *Získatelnost* – vybranou vlastnost lze měřit.
- *Výkon* – sytém dokáže vybranou vlastnost zpracovat dostatečně přesně a rychle.
- *Akceptace* – uživatelé jsou ochotni poskytnout vybranou vlastnost systému.
- *Odolnost vůči podvrhu* – daná vlastnost není jednoduše padělatelná.

2.2.1 Otisk prstu

Každý člověk má na špičce každého svého prstu tzv. papilární linie. Jedná se o unikátní struktury v kůži, které umožňují identifikaci jedinců, neboť podle tzv. daktyloskopických zákonů na světě neexistují dva lidé, kteří mají naprosto stejný obraz papilárních linií. Grafickou reprezentací papilárních linií jsou otisky prstů. V oblasti biometrie je rozpoznávání pomocí otisku prstu v současnosti nejrozšířenější technologií. Obor rozpoznávání otisků prstů pro kriminalistické účely se nazývá daktyloskopie a existuje již více než sto let. [3, s. 95] V komerční sféře využívají otisky prstů například některé americké supermarkety obchodního řetězce Kroger [2, s. 13] nebo laptopy a nejmodernější chytré telefony, jako je Apple iPhone 5S nebo Samsung Galaxy S5. [8] Otisky prstů jsou také součástí biometrických pasů České republiky. [9]

Tato metoda nebude v diplomové práci implementována, neboť jen nejnovější chytré telefony obsahují hardware pro snímání otisku prstu a aplikace by tak ve starších zařízeních vůbec nebyla nepoužitelná.

[8] Mezi další kritéria řadíme cenu, přijatelnost cílovými uživateli, typ použitého senzoru nebo vhodnost vybrané biometrické metody pro konkrétní typ aplikace. [2, s. 12]

2.2.2 Obličej

Velice známou biometrickou charakteristikou je lidský obličej. V běžném životě rozpoznáváme osoby na základě jejich obličeje zcela intuitivně. Automatizované rozpoznávání však není vůbec triviální úlohou, neboť obličej může vykazovat velice vysokou vnitrotřídní[9] variabilitu a v některých případech naopak nízkou mezitřídní[10] variabilitu. Biometrické systémy využívající rozpoznávání obličeje mohou být navíc v některých případech poměrně snadno podvedeny, a to např. předložením fotografie namísto skutečného obličeje. V současné době se kromě 2D snímků používají také 3D modely obličeje nebo termosnímky. [3, s. 153-54] V praxi nachází rozpoznávání pomocí obličeje široké spektrum využití, a to např. na letištích, při hraničních kontrolách, v docházkových a přístupových systémech nebo při policejních postupech (rozpoznávání osob při velkých kulturních nebo sportovních akcích). [3, s. 176-77] Fotografie obličeje je nedílnou součástí téměř jakéhokoliv identifikačního dokladu (cestovní pas, občanský průkaz, řidičský průkaz…).

Implementace uzamykací obrazovky s rozpoznáváním obličeje v diplomové práci realizována nebude. Tato metoda vyžaduje velice sofistikované řešení vzhledem k výše popsaným problémům, navíc v prostředí Android již řešení existuje (viz kapitola 3.1.4).

2.2.3 Duhovka

Duhovka je silně pigmentovaný orgán v lidském oku, v jehož středu se nachází zornice. Duhovka zajišťuje, že světelné paprsky vnikají do oka pouze zornicí a reguluje množství těchto paprsků. [10] Podle [3, s. 179] je vzorkování duhovky pro každého jedince zcela unikátní, což platí i pro jednovaječná dvojčata, přestože je barva i struktura duhovky dána geneticky. Duhovka je vnitřní orgán, který je proti poškození chráněn rohovkou, vzorkování duhovky je navíc neměnné v čase, díky čemuž je velice vhodnou biometrickou charakteristikou. Dle [11] je rozpoznávání na základě duhovky nejpřesnější biometrickou metodou. Jedná se o poměrně mladou technologii, která byla patentována až v roce 1994. Hlavní využití nachází jako náhrada cestovních dokladů na mezinárodních letištích[11], prostředek pro zabraňování vstupu nepovolaných osob do zón s omezeným přístupem nebo také v nemocnicích (např. párování matky a dítěte v porodnici).

Pro implementaci v diplomové práci není tato metoda příliš vhodná, protože zahrnuje náročné postupy zpracování obrazu, které by mohly příliš zpomalovat běh aplikace. Pro další práci s obrazem duhovky je navíc potřeba kvalitní snímek, pro jehož pořízení nemá stále ještě podstatné množství levnějších chytrých telefonů dostatečně výkonný fotoaparát. Mnoho telefonů také nedisponuje předním fotoaparátem a pořízení snímku vlastního oka by tak nebylo uživatelsky pohodlné.

---

[9] Způsobeno gestikulací, mimikou, stárnutím, ale i účesem, vousy, brýlemi nebo dalšími doplňky.
[10] Dvojčata, dvojníci.
[11] Např. letiště Schiphol v Amsterdamu využívá skenování duhovky pro urychlení imigračního procesu a také pro verifikaci identity zaměstnanců. [2, s. 12-13]

2.2.4 Chůze

Chůzi řadíme mezi behaviorální biometrické charakteristiky. Je vhodnou metodou při rozpoznávání osob na větší vzdálenosti, kdy je velice obtížné nebo i nemožné použít jiné biometrické údaje. Algoritmy pro ověřování identity na základě chůze berou v potaz statické (tvar těla) i dynamické znaky pohybujícího se těla. Mezi výhody této metody patří fakt, že chůzi lze snímat i bez vědomí pozorované osoby a klíčové znaky lze získat i ze záznamu s nízkým rozlišením. Hlavní nevýhodou je vysoká složitost, neboť je potřeba pracovat s video sekvencí, což je časově náročnější než práce s jediným snímkem. Lidská chůze je navíc ovlivňována velkým množstvím faktorů, jako je obuv, oblečení, rychlost, povrch, po kterém člověk kráčí atd. [4, s. 182-85] Rozpoznávání chůze je stále ve vývoji a zatím nelze nalézt širší využití v komerční sféře. V kriminalistice bývá tato metoda používána k identifikaci pachatelů na záznamu bezpečnostních kamer, užití nalézá i v medicíně pro odhalení Parkinsonovy choroby nebo roztroušené sklerózy v jejich raných stadiích. [12]

Ověřování identity na základě lidské chůze je pro implementaci v diplomové práci naprosto nevhodné. Pořízení video záznamu vlastní chůze pro odemknutí telefonu je pro uživatele nepřijatelné, obtížně realizovatelné a i v případě překonání tohoto omezení je tato metoda pro implementaci na mobilních zařízeních výpočetně příliš náročná.

2.2.5 Geometrie ucha

Podle [2, s. 131] objevil možnosti využití lidského ucha jako biometrické charakteristiky francouzský kriminalista Alphonse Bertillon již v roce 1890. Zatímco pro ostatní metody založené na rysech obličeje jsou potřebné snímky pořízené zepředu, geometrii ucha lze extrahovat ze snímku pořízeného z profilu. To může být v mnoha případech užitečné a lze tak použít geometrii ucha v případech, kdy není možná identifikace na základě obličeje. Ucho jako biometrika má několik pozitivních vlastností – struktura ucha zůstává během stárnutí stálá, vývoj ucha je dokončen ve věku čtyř let, dále již roste téměř lineárně a jeho proporce se nemění. Na rozdíl od ostatních znaků v obličeji se ucho nemění s výrazem tváře. Získávání snímku ucha navíc nevyžaduje aktivní interakci se senzorem. [4, s. 176]

K realizaci uzamykací obrazovky v diplomové práci by mohla být tato metoda vhodným kandidátem vzhledem k její relativní jednoduchosti. Z uživatelského pohledu ale není příliš komfortní snažit se pořídit snímek vlastního ucha, proto ověřování na základě geometrie ucha nebude v práci implementováno.

2.2.6 Dynamika úderů na klávesnici

Předpokládá se, že každý člověk píše na klávesnici svým charakteristickým způsobem. Tento způsob sice nebude unikátní napříč celou populací, ale přesto může poskytnout užitečné informace o uživateli při identifikačním procesu. Jedná se o behaviorální biometrickou charakteristiku s vysokou vnitrotřídní variabilitou, která je způsobena různými typy klávesnice, fyzickým nebo i duševním stavem uživatele apod. [4, s. 33] Historickým předchůdcem této metody je tzv. metoda

„*Fist of the Sender*“ z období 2. světové války, která sloužila k identifikaci telegrafistů podle dynamiky úderů při posílání zprávy Morseovou abecedou. [3, s. 245]

Ověřování identity na základě dynamiky psaní není samo o sobě příliš silným a spolehlivým bezpečnostním prvkem, nicméně jej lze použít jako doplněk klasického hesla nebo PIN kódu a razantně tak zvýšit bezpečnost těchto metod. Výhodou je, že tato biometrická metoda nemá žádné speciální nároky na hardware, senzorový modul zde tvoří klávesnice, případně dotykový displej. Uživatelé také nemusí explicitně poskytovat své biometrické údaje, jak tomu je u množství ostatních metod, ale mohou být ověřováni během svých přirozených akcí. Výpočet frekvence úderů na klávesnici nebude náročný ani pro limitovaná Android zařízení, tato metoda je tedy vhodná pro implementaci v rámci diplomové práce. Podrobněji je metoda popsána v kapitole 4.1.

2.2.7 Hlas

Lidský hlas řadíme na rozmezí fyziologických a behaviorálních biometrik. Tato charakteristika je totiž dána na jedné straně tvarem hlasového ústrojí, na straně druhé pak artikulací, náladou nebo i zdravotním stavem daného jedince. [4, s. 33] Rozpoznávání podle hlasu je lidem přirozené, v biometrických systémech rozlišujeme podle [3, s. 204] dvě možnosti identifikace jedince – *závislé a nezávislé na textu*. V případě textově závislého rozpoznávání je potřeba, aby uživatel vyslovil předem stanovenou frázi. Tato fráze může být unikátní pro každého uživatele (vlastní heslo) nebo stejná pro všechny, kteří systém užívají. Výhodou takového přístupu je jednodušší realizace, nevýhodou naopak snížení pohodlí uživatelů. Textově nezávislá hlasová identifikace je mnohem náročnější na implementaci, ale poskytuje přirozenější způsob ověřování identity. Biometrický systém v tomto případě dokáže ověřovat totožnost uživatele nezávisle na vyřčené frázi, navíc v nutných případech i bez vědomí uživatele. Rozpoznávání uživatele na základě hlasu je vhodnou biometrickou metodou v těch případech, kdy nelze využít jiné metody, například při ověřování identity přes telefon.

Implementace rozpoznávání uživatele podle hlasu byla v rámci diplomové práci původně zvažována, konkrétně přicházelo v úvahu jednodušší, textově závislé rozpoznávání. Jako základ by bylo možné využít algoritmů dostupných v balíčku *„android.speech“,* které využívají Android zařízení pro rozpoznávání řeči. Algoritmy ovšem dokáží pouze rozeznávat slova a fráze, nikoliv identifikovat řečníka. Navíc podle [13] až do verze Android 4.1 vyžadují přístup na internet, neboť výpočty jsou prováděny na vzdáleném serveru, čímž je celý proces zpomalen. Další algoritmy pro samotné rozpoznávání charakteristických rysů v řeči by také značně zvýšily výpočetní náročnost, pro realizaci uzamykací obrazovky je však potřeba rychlejších metod. Také je potřeba zmínit, že lidská řeč vykazuje velice vysokou vnitrotřídní variabilitu, což by mohlo ve výsledku způsobovat příliš vysokou míru FRR. S ohledem na popsaná fakta nebude ověřování identity na základě hlasu v diplomové práci implementováno.

2.2.8 Geometrie ruky

Ověřování identity podle tvaru lidské ruky má delší historii než by se mohlo na první pohled zdát. Malby objevené ve francouzské jeskyni Chaveut jsou „podepsány“ otiskem ruky, přičemž jejich stáří je datováno radiokarbonovou metodou na 31 000 let. V roce 1858 Sir William Herschel přiměl svého obchodního partnera Rajyadhara Konai, aby otisknul svou ruku na sepisovanou smlouvu, a tím se se smlouvou jednoznačně svázal. Prvním komerčním řešením byl scanner firmy Identimation s názvem Identimat, který byl představen na začátku 70. let 20. století a byl vyráběn až do roku 1987. [2, s. 91] Podle [3, s. 139] je nejdéle běžícím biometrickým systémem Identimat v jídelnách Univerzity v Georgii. Systém je v provozu již od roku 1973 a je stále modernizován. Některá současná řešení nepracují pouze s 2D geometrií, ale používají trojrozměrný obraz ruky. Další praktická nasazení systémů orientovaných na geometrii ruky jsou podle [2, s. 96] např. zabezpečení na mezinárodním letišti v San Franciscu nebo jaderných elektrárnách v USA, během olympijských her v roce 1996 bylo použita geometrie ruky pro kontrolu vstupu do olympijské vesnice. Podle [4, s. 186] je tato metoda vhodnější spíše k verifikaci než identifikaci, neboť nevykazuje příliš vysokou mezitřídní variabilitu.

Vzhledem k tomu, že aplikace implementovaná v rámci diplomové práce bude pracovat ve verifikačním, nikoli identifikačním módu, je tato metoda vhodným kandidátem k realizaci. Pořízení snímku vlastní ruky je také pro většinu uživatelů Android zařízení jednoduché a přijatelné. Snímek navíc nemusí mít ani vysokou kvalitu, metoda nebere v potaz detailní znaky ruky (vrásnění kůže, žíly,…), ale pouze její geometrii, tedy siluetu. Extrahování klíčových znaků ze siluety a další měření by také nemělo být výpočetně příliš náročným úkolem. Z dostupných metod biometrického ověřování identity se geometrie ruky jeví jako jedna z nejpřijatelnějších k implementaci na Android zařízení, a bude proto realizována v rámci diplomové práce.

2.2.9 DNA

DNA neboli deoxyribonukleová kyselina je jednodimenzionální nositelka genetické informace všech organismů s výjimkou některých nebuněčných. Svou strukturou zadává program buňkám, a tím určuje vývoj a vlastnosti celého organismu. [14] DNA tvoří neměnný a unikátní kód organismu, který může sloužit k jednoznačnému určení identity jedince s výjimkou jednovaječných dvojčat, která mají shodnou DNA. Rozpoznávání na základě DNA našlo nejširší uplatnění ve forenzní oblasti, pro komerční a vládní využití není tato metoda příliš vhodná, a to ze tří důvodů. Za prvé je velice jednoduché získat vzorek DNA, který může být dále zneužit. Dále tato metoda naráží na problémy v automatickém rozpoznávání, současné postupy porovnávacího procesu vyžadují náročné chemické metody a expertní dovednosti. Posledním velkým problémem určování identity na základě DNA je otázka soukromí. Z DNA lze například vyčíst náchylnost jedince k určitým chorobám, což by v jistých případech mohlo vést k diskriminaci, kupříkladu v přijímacích řízeních. [6, s. 8]

Je zřejmé, že zabezpečení Android zařízení pomocí ověřování DNA je nejméně vhodnou možností ze všech uvedených. Jak bylo popsáno v předchozím odstavci, automatické rozpoznávání identity na základě DNA je velice složitý proces se spoustou problémů, proto tato metoda nebude v diplomové práci implementována.

3 Použité technologie

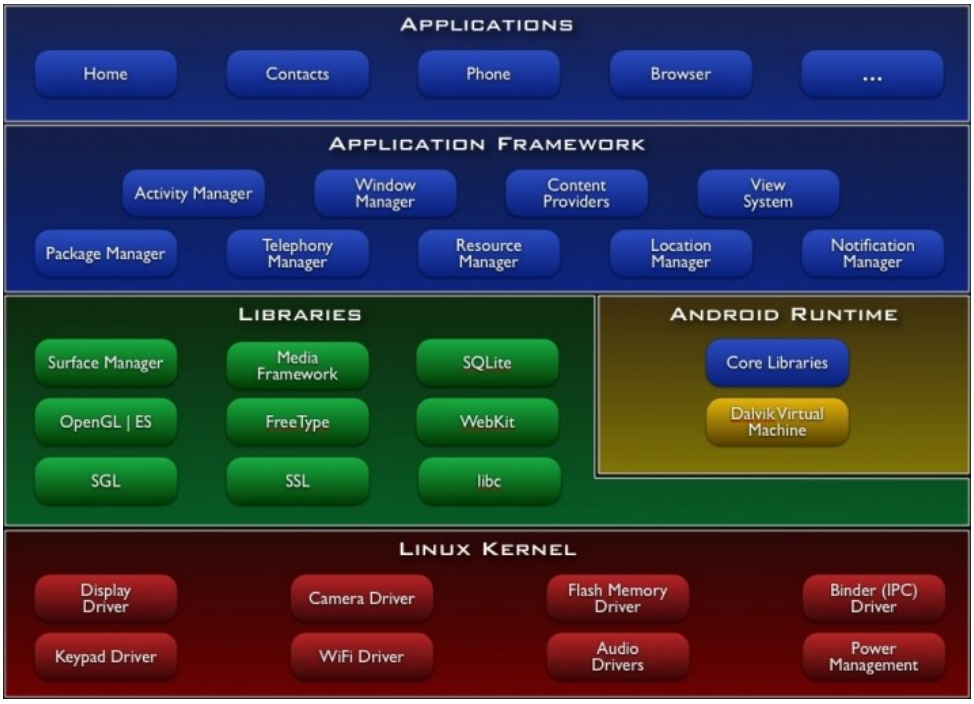
Kapitola popisuje technologie, s nimiž se bylo potřeba seznámit při psaní diplomové práce. Je zde představena platforma *Android* a popsán současný stav použití biometrik na této platformě. Dále je v kapitole prezentována knihovna pro práci s obrazovými daty – *OpenCV*.

3.1 Platforma Android

Android je rozsáhlá platforma s otevřeným zdrojovým kódem, která byla vytvořena společností Google. Je určen především pro chytré telefony a tablety. V současné době je vyvíjen sdružením více než osmdesáti firem z oblasti mobilních technologií s názvem *Open Handset Alliance* (dále jen OHA). Členy OHA jsou např. Samsung Electronics, HTC, T-Mobile, Google, Motorola, LG aj. Android je instalován ve stovkách typů mobilních zařízení ve více než 190 zemích po celém světě. [15] [16]

3.1.1 Architektura

Platforma Android je dle [17, s. 19-21] složena z pěti vrstev, jejichž architekturu znázorňuje Obrázek 3-1.



Obrázek 3-1: Architektura platformy Android, převzato z [18]

Nejnižší vrstva představuje jádro operačního systému a nazývá se **Linux Kernel.** Stará se o abstrakci mezi hardwarem a softwarem, koordinaci všech běžících procesů a podporuje správu paměti.

Vrstva s názvem **Libraries** (česky knihovny) obsahuje řadu API (Application Programming Interface) pro vývoj aplikací. Patří zde např. rozhraní *„android.util“* obsahující základní programovací nástroje, *„android.widget“* pro přístup k prvkům uživatelského rozhraní nebo *„com.google.android.maps“* pro přístup k mapám v aplikaci Google Maps.

Android runtime je vrstva, která obsahuje základní knihovny jazyka Java a virtuální stroj *Dalvik Virtual Machine* (dále jen DVM). DVM je optimalizovaný pro potřeby mobilních zařízení. Při programování Android aplikací se používá jazyk Java, který je následně přeložen do Java byte kódu a poté do mezikódu za pomoci Dalvik kompilátoru. Nakonec je mezikód spuštěn na DVM.

Nejdůležitější vrstvou z pohledu Android vývojáře je **Application Framework.** Tato vrstva umožňuje vývojářům přistupovat k různým službám, kontaktům, kalendáři, používat hardware zařízení, spouštět jiné aplikace atp.

Nejvyšší vrstvou v architektuře je vrstva s názvem **Applications** zahrnující samotné Android aplikace, které využívají uživatelé zařízení.

3.1.2 Stavební kameny Android aplikace

Android aplikace jsou psány v programovacím jazyce Java. Pro distribuci na Android zařízení je kód společně s dalšími daty a zdroji zkompilován do APK[12] souboru. Jedná se o archiv s příponou *„.apk“*, který obsahuje vše potřebné k tomu, aby byla aplikace nainstalována do Android zařízení. Aplikace se může skládat z různých komponent, které tvoří její základní stavební kameny. [19] Rozlišujeme čtyři základní typy komponent, následuje jejich popis podle [17, s. 39-41]:

Activity (Aktivita)

Aktivita je základní komponenta reprezentující samostatnou obrazovku s uživatelským rozhraním, která zpravidla poskytuje hlavní funkcionalitu aplikace. Většinou se aplikace skládá z více aktivit, které jsou mezi sebou provázány. Každá aktivita může spouštět další aktivity, v takovém případě je její činnost pozastavena systémem a aktivita je uložena do zásobníku. Zásobník je typu *„last in, first out“* (dále jen LIFO), takže pokud uživatel v aplikaci použije tlačítko „Zpět“, je obnovena činnost poslední vložené aktivity.

Service (Služba)

Služby v aplikaci provádí operace na pozadí a na rozdíl od aktivit neposkytují uživatelské rozhraní. Služba může být spuštěna jinou komponentou aplikace a její činnost může pokračovat i v případě, že uživatel spustí jinou aplikaci. Služba může například na pozadí přehrávat hudbu nebo stahovat data z internetu, aniž by blokovala činnost uživatele v jiných aplikacích.

[12] Android package

Content Provider (Poskytovatel obsahu)

Poskytovatel obsahu představuje způsob, jakým aplikace pracuje s daty. Systémoví poskytovatelé umožňují aplikaci přístup k obrázkům, zvukům, videím, kontaktům apod. Vývojář také může napsat vlastního poskytovatele obsahu, který povoluje nebo naopak znemožňuje sdílet aplikační data s dalšími aplikacemi.

Broadcast Receiver (Přijímač)

Komponenty typu „přijímač“ naslouchají systémovým nebo aplikačním oznámením a jsou odpovědné za patřičné reakce podle typu oznámení. Příkladem použití může být reakce na nízký stav baterie nebo příchozí hovor. Přijímač by neměl vykonávat mnoho práce, ale pouze předat řízení dalším komponentám, které dále řeší nastalou situaci.

Android systém je navržen tak, že každá aplikace může spustit komponentu jiné aplikace. Vzhledem k tomu, že každá aplikace běží v samostatném procesu s omezeným přístupem k ostatním aplikacím, nelze cizí komponentu spustit přímo, ale za pomoci systému. Aktivity, služby a přijímače mohou být aktivovány pomocí asynchronní zprávy nazývané *intent*, česky *záměr*. Tato zpráva je reprezentována objektem s názvem **Intent**, který systému sděluje, kterou operaci je potřeba vykonat. [15]

Důležitou součástí každé Android aplikace je tzv. **Android Manifest**. Jedná se o xml soubor, který specifikuje parametry a požadavky (např. vyžadovaný hardware) pro chod aplikace a sděluje systému, z jakých komponent je aplikace složena. Manifest také obsahuje seznam oprávnění nutných pro fungování aplikace, např. přístup ke kontaktům, datům na externím uložišti, vypnutí zámku obrazovky, přístup k polohovým datům a další. Všechna oprávnění musí být schválena uživatelem během instalace aplikace. [17, s. 45]

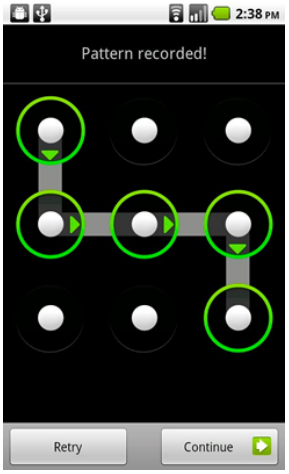
3.1.3 Uzamykací obrazovky

Tato podkapitola popisuje možnosti zabezpečení mobilního zařízení pomocí různých typů uzamykacích obrazovek. Uzamykací obrazovka může pomoci chránit data v mobilním zařízení lépe než například PIN kód na SIM kartě, který je vyžadován pouze při zapínání zařízení. Oproti tomu uzamykací obrazovka je aktivována již po chvíli nečinnosti zařízení, čímž dokáže zařízení lépe chránit v běžném životě.

Pravděpodobně nejpoužívanější uzamykací obrazovkou je tzv. **slide**. Pro odemknutí telefonu stačí pouze přejet prstem po obrazovce z bodu A do bodu B, což je určitě nejrychlejší a nejpohodlnější způsob odemykání. Bohužel ale poskytuje nejnižší úroveň zabezpečení – chrání zařízení pouze před nechtěnými akcemi v kapse uživatele, nikoliv před přístupem neoprávněných uživatelů.

Zajímavým typem uzamykací obrazovky je tzv. **pattern**, česky vzor. Uživatel svůj vzor nastaví libovolným propojením až devíti bodů na obrazovce, čímž vznikne vzorová křivka (viz Obrázek 3-2). Pro odemknutí zařízení je potřeba na displeji nakreslit stejnou křivku. Tato metoda zvyšuje

zabezpečení zařízení, ale i takový zámek lze prolomit. Kupříkladu pokud uživatel pravidelně nečistí displej svého zařízení, je teoreticky možné na displeji vyčíst stopy jeho vzorové křivky.



Obrázek 3-2: Příklad uzamykací obrazovky typu „pattern“, převzato z [20]

Od verze Android 4.0 (Ice Cream Sandwich) je v nabídce také biometrická metoda uzamykací obrazovky, konkrétně lze k odemčení zařízení využít vlastní **obličej.** K zachycení snímku obličeje se používá přední kamera zařízení. [21] Zámek lze však jednoduše prolomit podstrčením fotografie uživatele. Z toho důvodu byla ve verzi Android 4.1 (Jelly Bean) přidána kontrola živosti – uživatel musí při ověřování mrknout, aby bylo zabráněno odemykání zařízení pomocí fotografie.

Poslední dvě možnosti, které Android standardně nabízí, je uzamknutí zařízení pomocí numerického **PIN** kódu nebo alfanumerického **hesla**. V rámci klasických zabezpečovacích metod se tyto dvě stále jeví jako nejbezpečnější, nicméně ne příliš pohodlné. Zvláště pak pro uživatele, který své zařízení používá často během dne. Vyšší úroveň zabezpečení mohou poskytnout biometrické metody, které jsou popsány v kapitole 3.1.4.

3.1.4 Android a biometriky

Využití biometrických charakteristik pro zabezpečení mobilního zařízení s operačním systémem Android v sobě má veliký potenciál, který ještě není naplno využit. Tato podkapitola uvádí přehled aplikací a biometrických metod, které jsou v prostředí Android implementovány.

Pokud člověk hledá na Google Play[13] aplikaci pod klíčovým slovem *„biometrics“,* dostane výsledkem zástup aplikací, které slibují zabezpečení zařízení pomocí otisku prstu. Zde je potřeba zdůraznit, že drtivá většina těchto aplikací neposkytuje skutečnou biometrickou ochranu dat. Tyto aplikace

[13] Google Play je náhrada za dřívější Android Market; jedná se v podstatě o internetový obchod, kde si uživatelé Android zařízení mimo jiné kupují nebo stahují aplikace do svého zařízení. [40]

v sobě většinou skrývají jednoduchý trik pro odemknutí obrazovky, který zná jen uživatel, a pouze se tváří jako opravdové čtečky otisku prstů, což ale pro odrazení potenciálního útočníka může někdy stačit. Mezi takové aplikace patří např. *Finger Scanner Lite, Finger Scanner Lock, Fingerprint Lock Free* aj. Ve skutečnosti není možné skenovat otisky přiložením prstu na displej, používání podobných aplikací je tedy spíše otázkou zábavy než bezpečnosti. [22] [23]

Aplikace, které pracují se skutečnými otisky prstů, vyžadují speciální hardware, který umožní otisky prstů nasnímat. V současnosti není na trhu příliš mnoho zařízení, které by byly takovým hardware vybaveny. V dubnu 2014 se na Google Play objevila aplikace s názvem *ICE Unlock Fingerprint Scanner,* která pro získání otisku prstu používá fotoaparát zařízení. Je zřejmé, že snímek musí mít dostatečně vysoké rozlišení a ne každé zařízení disponuje tak kvalitním fotoaparátem. Mezi zástupce zařízení se senzorem otisku prstů patří Motorola Atrix a Samsung Galaxy S5. Pro chytrý telefon Motorola Atrix existuje aplikace s názvem *DataDefender*, která využívá snímač otisků prstů vestavěný v zařízení. [23] Samsung Galaxy S5 má ve svém domovském tlačítku také zabudovánu čtečku otisku prstů, navíc Samsung ke své čtečce vydává také speciální API, takže její funkce mohou využívat i vývojáři aplikací třetích stran. [8]

Kromě uzamykacích obrazovek samozřejmě existují také další aplikace biometrického ověřování identity na Android zařízení. Současní výrobci se orientují především na oblast bezpečnosti v mobilních zařízeních v souvislosti s používáním online služeb. Jedná se např. o přístup k bankovním aplikacím, emailovým a dalším účtům. Aliance *Fast IDentity Online* (dále jen FIDO[14]) pro tyto účely vytvořila vlastní čtečku otisku prstů, kterou od roku 2014 plánuje instalovat do množství nových Android zařízení. [24] Podle [25] společnosti *BIO-Key* a *InterDigital* společně pracují na autentizačním řešení založeném na otiscích prstů, které poskytuje nejen uzavřené ověřování v zařízení, ale právě také ověřování identity v rámci sítě.

Ověřování identity pomocí otisku prstů je v Android zařízeních zřejmě nejpopulárnější metodou, nicméně na trhu lze nalézt i řešení využívající jiné biometrické charakteristiky. Jak bylo popsáno v podkapitole 3.1.3, od verze Android 4.0 je k dispozici uzamykací obrazovka s ověřováním identity podle obličeje. Tento způsob odemykání je pro uživatele atraktivní a pohodlný, nicméně metoda má stále ještě bezpečnostní trhliny. Původně bylo možné systém obelstít předložením fotografie, problémy s bezpečností úplně nevyřešilo ani přidání povinnosti mrknout, jelikož není příliš složité vytvořit z fotografie krátkou animaci, která mrknutí napodobuje. Z dalších výrobců stojí za zmínku např. společnost *Mobbeel*, která vyvíjí řešení pro Android kombinující duhovku, hlas, obličej nebo podpis. [26] Aktuálně lze na Google Play nalézt její aplikaci *BioWallet*, která umožňuje chránit data v zařízení pomocí vlastnoručního podpisu. Společnost *Fulcrum Biometrics* poskytuje vlastní

[14] Konsorcium společností, jehož cílem je zavést rychlejší a bezpečnější online ověřování identity (od hesel směrem k biometrikám a dalším metodám). Členy jsou například PayPal, Lenovo, Google, BlackBerry, MasterCard, Microsoft aj. [36] [37]

biometrické řešení pomocí porovnávání obličeje, duhovky a otisku prstů. Aplikace s názvem *Secure BRaVe App* však vyžaduje externí senzor pro snímání otisků prstů i pro oční duhovku. [27]

3.2 OpenCV

Vzhledem k tomu, že aplikace implementovaná v diplomové práci využívá i náročnější algoritmy zpracování obrazu, bude nutné vhodně zvolit přístup ke zpracování obrazových dat. První možností by bylo psát algoritmy přímo v jazyce Java, pro práci s obrazem poskytuje Android balíček s názvem *„android.graphics“*. Pro náročnější operace se ale nejedná o optimální řešení s ohledem na výkon aplikace. Druhou možností je využití existující knihovny; nejlépe zdokumentovanou knihovnou pro zpracování obrazu je *OpenCV*.

OpenCV je knihovna s otevřeným zdrojovým kódem, která je nativně psána v jazyce C++ a obsahuje více než 2500 optimalizovaných algoritmů pro práci s obrazem. Algoritmy lze s úspěchem využívat pro detekci obličeje, objektů a hran, aplikaci nejrůznějších filtrů a mnohé další výpočty v obraze. Knihovna poskytuje rozhraní pro jazyky C, C++, Python, Java a MATLAB a funguje na platformách Windows, Linux, Android a Mac OS. OpenCV je publikována pod licencí BSD, která patří mezi nejsvobodnější [28] a umožňuje široké použití i v komerční sféře. Služeb knihovny využívají firmy jako Google, Yahoo, Microsoft, Intel, Sony, Honda, Toyota a mnohé další. [29]

4 Volba vhodných biometrických metod

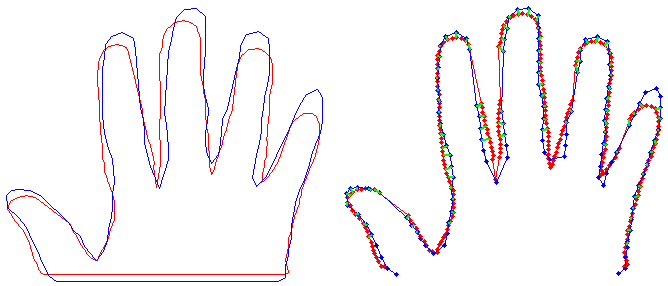
Kapitola se věnuje bližšímu rozboru biometrických metod, které byly zvoleny k implementaci v diplomové práci. Je zde vysvětleno, jaké postupy se obecně používají k realizaci zvolených metod a vybrán konkrétní způsob, jak budou metody řešeny v rámci diplomové práce. Volba metod k implementaci závisela na několika faktorech. Cílem bylo vybrat zástupce statických i dynamických metod ověřování, a to nejlépe takových, které ještě nejsou v prostředí Android implementovány. Důležitým rozhodovacím kritériem byla použitelnost dané metody na co možná nejširším spektru mobilních Android zařízení. Proto nejsou k implementaci vybrány žádné z metod vyžadující speciální hardware, jako jsou např. ověřování na základě otisku prstu nebo duhovky. Detailnější diskuze o vhodnosti různých biometrických metod je uvedena v rámci kapitoly 2.2 u popisu jednotlivých metod ověřování.

4.1 Geometrie ruky

Senzory pro snímání geometrie ruky mohou získávat 2D nebo 3D obrazy. Pro vytvoření 3D snímků je však potřeba speciálního zařízení, které dokáže ruku snímat z různých úhlů, a získat tak několik dvojrozměrných snímků ruky. Následně je z těchto snímků sestaven 3D obraz, který obsahuje větší množství biometrických údajů a tím poskytuje vyšší úroveň zabezpečení. V zamýšlené aplikaci však bude k dispozici pro snímání ruky pouze jeden fotoaparát, který je součástí Android zařízení, proto zde budeme pracovat pouze s dvojrozměrnými snímky.

Ke zpracování 2D obrazu ruky lze přistoupit několika způsoby. Podle [3, s. 128] existuje *metoda zarovnání rukou, metoda analýzy šířky prstů a metoda založená na přímých měřeních*. U každé metody je prvním krokem extrakce obrysu ruky ze snímku, k čemuž se používá adaptivní prahování neboli binarizace.

Metoda založená na zarovnání rukou pracuje na principu, že aktuální obraz je natočen stejným způsobem jako šablona. Oba obrazy jsou poté položeny tak, aby se překrývaly, a počítá se skóre odlišnosti obou vzorků (viz Obrázek 4-1).



Obrázek 4-1: Zarovnání rukou, převzato z [30]

Metoda založená na analýze šířky prstů pracuje tak, že je nalezena hlavní a vedlejší osa ruky. Vedlejší osa rozděluje ruku na oblast zápěstí a prstů. V bodě, kde se obě osy protínají, začne algoritmus postupovat po obrysu ruky a hledá lokální minima a maxima, které značí špičky prstů a údolí mezi nimi. Tím je obraz rozdělen na jednotlivé prsty a každý prst je dále samostatně zpracováván. Každý bod je z okraje prstu promítnut na osu prstu a je vytvořen histogram vypočítaných vzdáleností. Následně se počítá pravděpodobnostní rozložení všech prstů, které je porovnáváno se šablonou pro určení podobnosti.

Metoda založená na přímých měřeních používá standardních vlastností ruky (jako je šířka dlaně, délky prstů apod.) k porovnávání podobnosti mezi šablonou a aktuálním snímkem. Aby bylo zajištěno, že ruka bude vždy nasnímána ve stejné poloze, lze umístit na senzor distanční kolíky určující uživateli, jak ruku položit (viz Obrázek 4-2). Tím je usnadněna a zrychlena práce algoritmu, který se již nemusí starat o korekce orientace ruky. Algoritmus poté stejně jako u předchozí metody hledá špičky prstů a údolí mezi nimi, případně další významné body na obrysu ruky a počítá vzdálenosti mezi nalezenými body. Vzdálenosti jsou uloženy ve formě vektoru. Podobnost šablony a aktuálního vzorku je počítána jako absolutní, váhová nebo Euklidovská vzdálenost mezi danými vektory.

Obrázek 4-2: Distanční kolíky na scanneru ruky, převzato z [31]

V implementované aplikaci bude použita kombinace postupů popsaných metod tak, aby práce algoritmu byla co nejjednodušší a nejrychlejší. Vzhledem k povaze aplikace je nepřípustné používat externích podložek nebo jiných zařízení s distančními kolíky pro správné umístění a snímání ruky. K získání snímku bude používán fotoaparát Android zařízení. Na displeji zařízení může být vykreslena silueta ruky tak, aby uživatel věděl, jakým způsobem svou ruku nafotit. Obraz ruky bude určitě dále binarizován a obrys procházen pro nalezení špiček prstů a údolí mezi prsty. Nejrychlejším přístupem se jeví počítání vzdáleností mezi těmito významnými body, ukládání vzdáleností ve formě vektoru a počítání podobnosti mezi aktuálním vzorkem a šablonou pomocí Euklidovské vzdálenosti.

4.2 Dynamika úderů na dotykovém displeji

V současné době drtivá většina přístrojů s operačním systémem Android nepoužívá jako vstupní zařízení klávesnici, ale dotykový displej. Proto druhá biometrická metoda implementovaná v rámci

diplomové práce bude variací tradiční metody analýzy dynamiky úderů při psaní na klávesnici adaptované pro dotykový displej. K ověřování identity uživatele pomocí dynamiky psaní na klávesnici lze přistoupit dvěma způsoby. První možností je uživatele průběžně monitorovat při jeho práci. Pokud dojde k náhlé změně v jeho způsobu psaní, lze předpokládat, že se k počítači dostal neoprávněný uživatel a podniknout dle toho příslušná opatření. Druhou možností je statické ověřování identity, kdy je uživatel nucen napsat heslo nebo dohodnutou frázi a systém zkoumá způsob zadání vybrané fráze. Při používání této metody na klávesnici je dynamika ovlivněna také konkrétní psanou frází z důvodu rozložení jednotlivých kláves.

U dotykového displeje tento problém odpadá, neboť celá plocha displeje se může chovat jako jediné tlačítko a uživatel tak může klepat na jediné místo. Pro potřeby uzamykací obrazovky je vhodnější statická metoda, kdy bude uživatel nucen poklepat na zařízení během stanoveného časového rozmezí, případně bude mít určen počet klepnutí. Systém bude následně měřit intervaly mezi jednotlivými klepnutími a zkoumat, zda se počet klepnutí a intervaly mezi nimi shodují se šablonou. Způsob implementace může být inspirován metodou *„Fist of the Sender“*, která již byla zmíněna v podkapitole 2.2.6.

5 Analýza a návrh

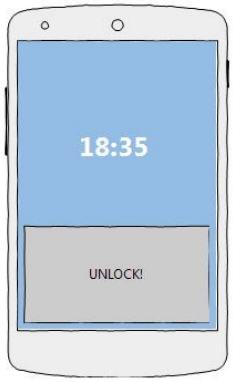
Kapitola demonstruje náhledy uživatelského rozhraní pro jednotlivé obrazovky vyvíjené aplikace a také blíže popisuje, jaká kritéria by měla aplikace splňovat, aby mohla být považována za zámek obrazovky. Je zde vysvětleno rozvržení aplikace na komponenty a přiblížena funkcionalita jednotlivých komponent.

5.1 Specifika uzamykací obrazovky

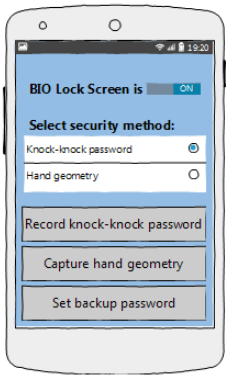
Uzamykací obrazovku můžeme chápat jako aplikaci, která se zobrazí v celoobrazovkovém módu okamžitě po zapnutí displeje. Při aktivním zámku obrazovky by měl být uživateli skrytý i tzv. status bar (stavový řádek s informacemi o stavu baterie, aktivních připojeních apod.), který je standardně na zařízeních se systémem Android stále zobrazen, pokud aplikace nepracuje v zobrazení na celou obrazovku. Zámek obrazovky chrání telefon před nechtěnými akcemi (přístup na internet, volání, odesílání zpráv apod.) bez vědomí uživatele, například v kapse nebo v tašce. Taková nevědomá interakce se zařízením může způsobit nemalé problémy kupříkladu při vyúčtování za užívané služby. Pokud uživatel používá jinou než základní obrazovku typu „slide“, poskytuje uzamykací obrazovka také jistou úroveň zabezpečení a chrání zařízení před neoprávněnými uživateli. Ukončit zobrazení zámku, a tím odblokovat zařízení k běžnému používání, lze jedním ze specifických způsobů popsaných v kapitole 3.1.3. Pokud je v popředí aplikace zámku obrazovky, je důležité, aby ji uživatel dokázal ukončit pouze stanoveným způsobem. Proto je potřeba, aby byla náležitě ošetřena reakce aplikace na hardwarová tlačítka, která většinou mají jasně danou funkčnost napříč všemi aplikacemi. Tlačítko „Domů“ zobrazuje domovskou obrazovku zařízení, tlačítko „Zpět“ vyvolá ze zásobníku poslední vloženou aktivitu, některá zařízení mají další hardwarová tlačítka například pro zobrazení nastavení aplikace. Důležité je, aby byl zámek schopen reagovat na příchozí hovory a umožnil uživateli přijmout hovor bez nutnosti odemknout obrazovku, nicméně po ukončení hovoru by měl být zámek stále aktivní. Uživatelsky přívětivé je také zobrazovat během uzamčení aspoň základní informace, např. aktuální čas.

5.2 Uživatelské rozhraní

Vyvíjená aplikace se bude skládat celkem z pěti různých obrazovek. První obrazovka bude představovat samotný zámek, bude tedy zobrazovat pouze aktuální čas a tlačítko pro odemknutí telefonu (Obrázek 5-1). Další obrazovka bude sloužit k nastavení aplikace a bude se jednat o hlavní obrazovku, která se zobrazí uživateli při spuštění aplikace (Obrázek 5-2). Uživatel zde bude mít možnost vybrat si, jakou biometrickou metodu chce používat, nastavit záložní heslo pro odemknutí telefonu v případě selhání biometrického ověřování, aktivovat zámek a vytvořit šablony svých biometrických charakteristik.

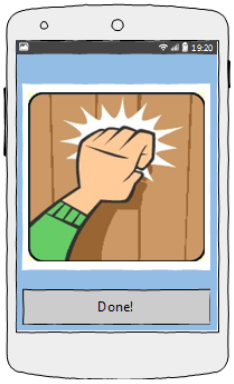


Obrázek 5-1: Hlavní uzamykací obrazovka



Obrázek 5-2: Obrazovka nastavení aplikace

Další dvě obrazovky využije uživatel pro snímání geometrie své ruky (Obrázek 5-4), respektive svého klepání na displej zařízení (Obrázek 5-3). Tyto obrazovky budou stejné jak v registrační fázi, kdy uživatel nahrává své údaje pro účely vytvoření šablony, tak v ověřovací fázi, kdy bude uživatel muset poskytnout své údaje k odemknutí telefonu. Při snímání geometrie ruky uživatel na displeji uvidí siluetu ruky tak, aby věděl, jakým způsobem svou ruku nasnímat. Toto opatření do jisté míry nahrazuje distanční kolíky používané v hardwarových snímačích geometrie ruky.



Obrázek 5-3: Obrazovka nahrávání klepání

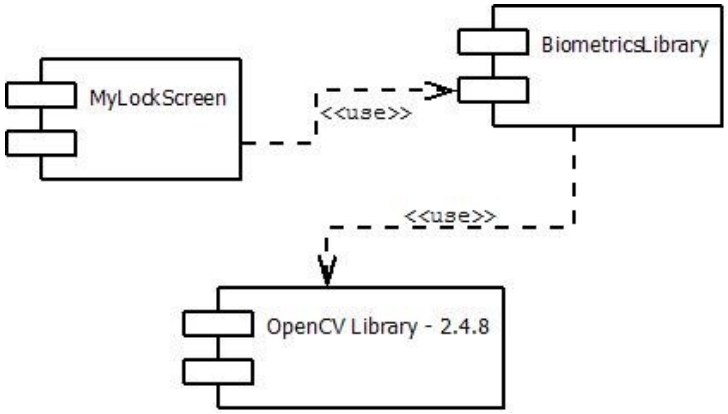
Obrázek 5-4: Obrazovka snímání ruky

Posledním důležitým bodem uživatelského rozhraní je obrazovka s náhledem zpracované geometrie ruky. Uživatel zde uvidí, zda aplikace správně detekovala významné body na obrysu jeho ruky, a bude mít možnost potvrdit nebo odmítnout uložení šablony do databáze (Obrázek 5-5).

Obrázek 5-5: Obrazovka náhledu geometrie ruky

5.3 Rozvržení projektu

Důležitým krokem při vývoji projektu je správná identifikace komponent a jejich odpovědností v rámci aplikace. Rozvržení projektu ukazuje diagram komponent, viz Obrázek 5-6. Je vhodné, aby od sebe byly odděleny části sloužící ke komunikaci s uživatelem a části poskytující funkce biometrického ověřování. Za tímto účelem je vytvořena knihovna obsahující biometrické metody (**BiometricsLibrary**) a samostatná aplikace zámku obrazovky (**MyLockScreen**), která funkcí knihovny využívá. Výhodou takového rozdělení je možnost znovupoužití biometrické knihovny v jiných aplikacích a naopak eventuálně aplikovat i jiné metody zámku obrazovky. Jak bylo zmíněno v kapitole 3.2, aplikace potřebuje zpracovávat obrazová data, proto některé metody biometrické knihovny využívají externí knihovnu **OpenCV Library – 2.4.8** (dále jen knihovna OpenCV).



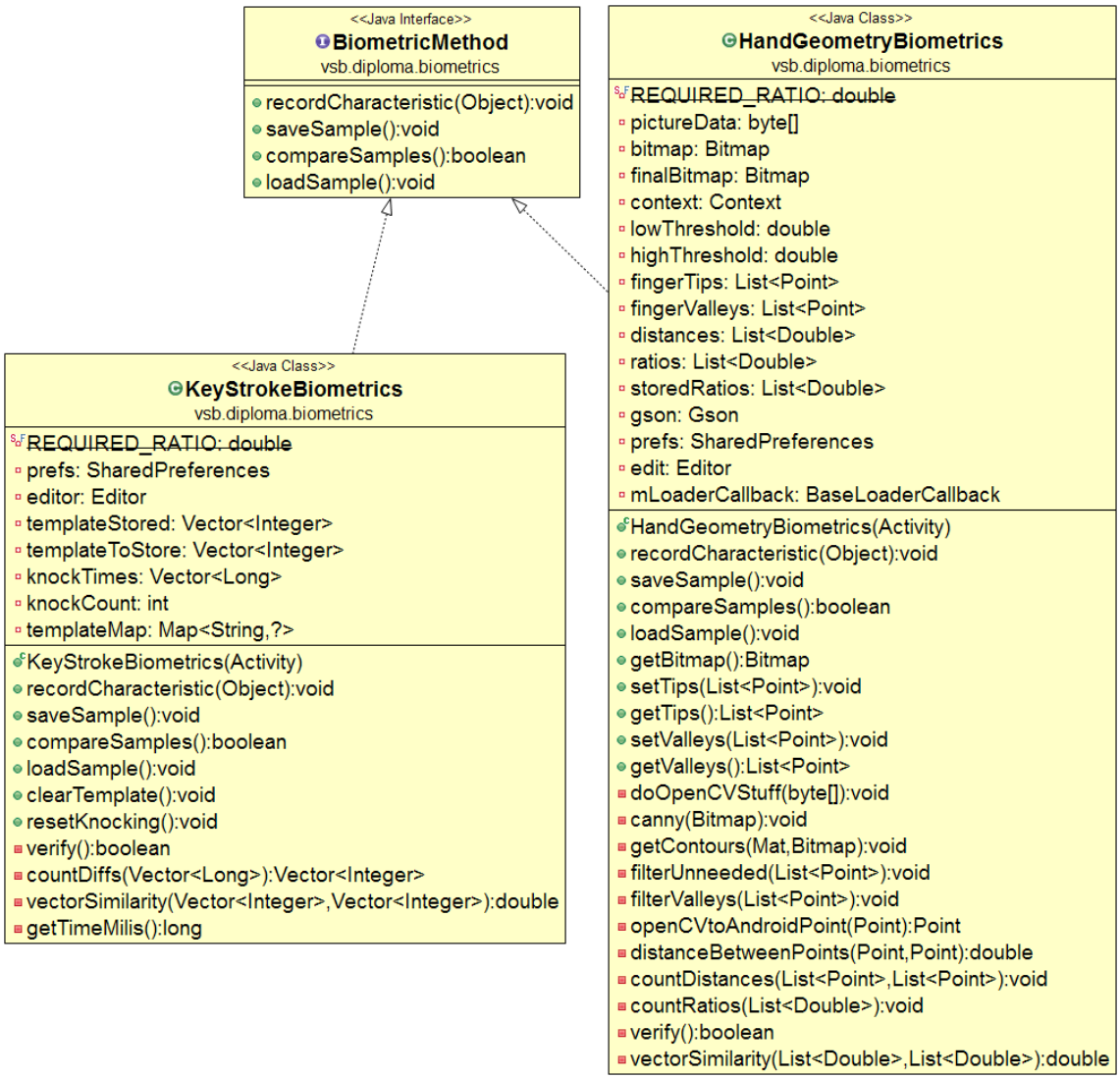
Obrázek 5-6: Diagram komponent projektu

6 Implementace

Kapitola popisuje konkrétní postupy při vývoji aplikace zámku obrazovky i knihovny, která bude zámkem využívána. Je vylíčena implementace zvolených biometrických metod i aplikace zámku obrazovky jako celku.

6.1 Tvorba biometrické knihovny

Dle zadání diplomové práce bylo potřeba vybrat dvě metody biometrického ověřování a implementovat je v podobě knihovny, která bude umožňovat přidávat další metody pro biometrickou verifikaci prostřednictvím definovaného rozhraní. Třídní diagram knihovny ukazuje Obrázek 6-1.



Obrázek 6-1: Třídní diagram knihovny BiometricsLibrary

Základ knihovny s názvem **BiometricsLibrary** tvoří rozhraní **BiometricMethod**. Rozhraní definuje základní metody pro práci s biometrickými charakteristikami, tedy *recordCharacteristic(Object data)* pro získání biometrických údajů od uživatele, *saveSample()* pro zpracování údajů a následné uložení vytvořené šablony, *compareSamples()* pro porovnání šablony a aktuálně poskytované biometrické charakteristiky a *loadSample()* pro načtení uložené šablony. Popsané metody musí být implementovány každou třídou reprezentující vybraný způsob biometrického ověřování, která bude obsažena v knihovně. Jako reprezentant statických metod byla pro implementaci vybrána geometrie ruky (třída **HandGeometryBiometrics** popsaná v podkapitole 6.1.1), z dynamických metod ověřování je implementována dynamika úderů na dotykovém displeji (třída **KeyStrokeBiometrics** popisovaná v podkapitole 6.1.2).

6.1.1 Geometrie ruky

Práci s biometrickými daty geometrie ruky zajišťuje třída **HandGeometryBiometrics**. Jak bylo popsáno v předchozím odstavci, třída implementuje rozhraní **BiometricMethod** a musí tedy implementovat všechny metody rozhraní. Metoda *recordCharacteristic(Object data)* v prvé řadě asynchronně načítá knihovnu **OpenCV** a dále slouží ke zpracování obrazu ruky uživatele. Jako parametr dostává obraz ve formě pole bytů, které si převede na datový typ Bitmap, a ten je dále zpracováván. V následujícím textu nebudou pro zjednodušení uvedeny všechny parametry všech používaných funkcí, ale pouze ty, které je vhodné popsat. Většina algoritmů v knihovně OpenCV totiž vyžaduje jako parametry mimo jiné také vstupní a výstupní obrazová data.

Bitmap je pomocí OpenCV algoritmů nejprve převeden na objekt typu Mat (matice), který je v knihovně OpenCV základním datovým typem reprezentujícím obrázek. S maticí mohou být prováděny nejrůznější transformace tak, aby následně tvořila co nejvhodnější vstup pro algoritmus detekování hran. Zde je důležité, aby byla ruka dobře rozlišitelná od pozadí, což není triviální úkol a pro každý snímek může být vhodná jiná úprava obrazu. Při vývoji aplikace bylo experimentováno s následujícími úpravami – převedení obrazu z barevného na černobílý, extrakce jedné z barevných složek RGB nebo převedení z RGB barevného modelu do modelu HSV a následná extrakce jedné z těchto složek (odstín, jas, sytost). Nejpřesnější výsledky poskytoval postup převedení obrazu do modelu HSV a extrakce jasové složky. Všechny popsané transformace mohou být jednoduše realizovány pomocí funkcí knihovny OpenCV.

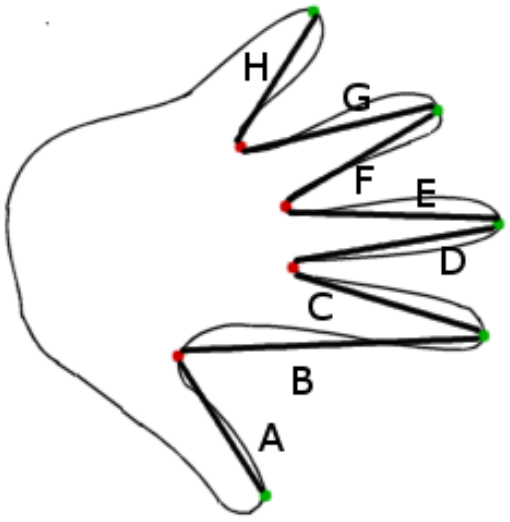
Upravená matice slouží jako vstup pro algoritmus detekce hran, který převede obraz do binární podoby. OpenCV nabízí k detekci hran dva různé algoritmy (Cannyho nebo Sobelův), během experimentů vykazoval lepší výsledky Cannyho algoritmus, proto je v práci využita funkce *Imgproc.Canny()*. Tato funkce vyžaduje mimo jiné dva parametry pro stanovení spodního a horního prahu. Tyto prahy jsou velice důležité pro správné rozlišení hran v obraze. Lze je nastavit libovolně, nicméně pro různé obrazy mohou být potřeba různé hodnoty prahů. Podle [32] lze hodnoty prahů spočítat automaticky tak, že je vypočtena střední hodnota odstínu šedi jednotlivých pixelů v obraze. Spodní práh je pak roven 0.66×(střední hodnota), horní práh se rovná 1.33×(střední hodnota). Střední hodnota je vypočtena pomocí OpenCV funkce *Core.mean()*.

Po detekci hran Cannyho algoritmem máme k dispozici binární obraz reprezentující hrany v obraze. Nyní předpokládáme, že nejdelší spojitá hrana tvoří obrys ruky. Pomocí OpenCV funkce *Imgproc.findContours()* získáme seznam (datová struktura List) všech hran. Každá položka seznamu je tvořena maticí bodů. Tento seznam je následně procházen a jsou sčítány body v jednotlivých položkách seznamu. Položka s největším počtem bodů je tedy hledaným obrysem ruky.

Poté, co je identifikována hrana tvořící obrys ruky, lze přistoupit k dalším krokům zpracování. Jsou procházeny jednotlivé body nalezené hrany a hledají se body, které jsou na špičkách prstů, respektive v údolích mezi prsty. Vzhledem k tomu, že je pevně daná poloha ruky v obraze (viz kapitoly 4.1 a 5.2), lze tyto body nalézt následovně: Postupujeme kupředu bod po bodu a porovnáváme x souřadnici sousedních bodů. X souřadnice následujícího bodu musí být vždy větší než x souřadnice bodu předchozího. V případě, že x souřadnice následujícího bodu je menší než x souřadnice bodu předchozího, je jasné, že bylo dosaženo vrcholu, tedy špičky prstu. Tento bod si tedy uložíme do vektoru špiček prstů a postupujeme dále po obryse ruky. Nyní je x souřadnice bodu vždy menší než x souřadnice bodu předchozího až do chvíle, kdy se x souřadnice začne opět zvětšovat, což je signálem, že bylo dosaženo údolí mezi dvěma prsty a daný bod je uložen do vektoru údolí. Takto je postupováno, dokud algoritmus neprojde všechny body hrany.

V této chvíli máme dva vektory bodů, které reprezentují špičky prstů a údolí mezi nimi. Z důvodu drobných nerovností v obryse ruky se stávalo, že algoritmus označil jako špičku nebo údolí více bodů, které byly blízko u sebe. Proto jsou oba vektory znovu procházeny a přebytečné body odfiltrovány tak, aby zbylo přesně pět špiček a čtyři údolí.

Metoda *countDistances()* počítá podle vztahu 6.1 vzdálenosti mezi špičkami a údolími a metoda *countRatios()* počítá poměry mezi vybranými vzdálenostmi (viz Obrázek 6-2; konkrétně jsou počítány poměry délky palce vůči délkám ostatních prstů, na obrázku tedy poměry délek A:B, A:C, A:D, A:E, A:F, A:G, A:H).



Obrázek 6-2: Významné vzdálenosti na geometrii ruky

Tyto poměry tvoří konečný biometrický markant a jsou uchovávány ve formě vektoru. Pomocí metody *saveSample()* může být vektor uložen do databáze. Persistentní ukládání dat v prostředí Android lze jednoduše realizovat pomocí rozhraní **SharedPreferences**, které umožňuje ukládat data ve formě klíč – hodnota.

$$vzdálenost\ bodů = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2} \tag{6.1}$$

kde A a B jsou body a x_A, y_A, x_B, y_B jsou jejich souřadnice.

Metoda *compareSamples()* porovnává aktuální data poskytovaná uživatelem se šablonou v databázi, kterou lze získat voláním metody *loadSample()*. Vstupem metody *compareSamples()* jsou tedy dva vektory reprezentující aktuální a uložené poměry vzdáleností mezi špičkami a údolími prstů. V prvním kroku metoda ověří, zda jsou stejné velikosti obou vektorů, tedy zda jí byly poskytnuty správně zpracované vektory. Pokud by se jejich velikost nerovnala, může to značit nekorektní vstupní data, porovnávání neproběhne a metoda vrátí negativní výsledek. Pokud je podmínka splněna, je počítána míra kosinové podobnosti vektorů podle vztahu 6.2. V případě, že je výsledná míra vyšší než nastavený práh (konstanta *REQUIRED_RATIO*), je výsledek porovnávání kladný.

$$míra\ podobnosti = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}} \tag{6.2}$$

kde A a B jsou vektory.

6.1.2 Dynamika úderů na dotykovém displeji

Postupy spojené se zpracováním úderů na dotykovém displeji zaštiťuje třída s názvem **KeyStrokeBiometrics**. Tato třída si udržuje vektor s názvem *knockTimes* a při snímání úderů na displeji metoda *recordCharacteristic(Object data)* přidává do vektoru časový údaj, kdy došlo ke klepnutí na displej. Pro získání potřebného časového údaje je volána systémová metoda *android.os.SystemClock.elapsedRealtime()*, která vrací počet milisekund, jež uběhly od zavedení systému (zapnutí zařízení).

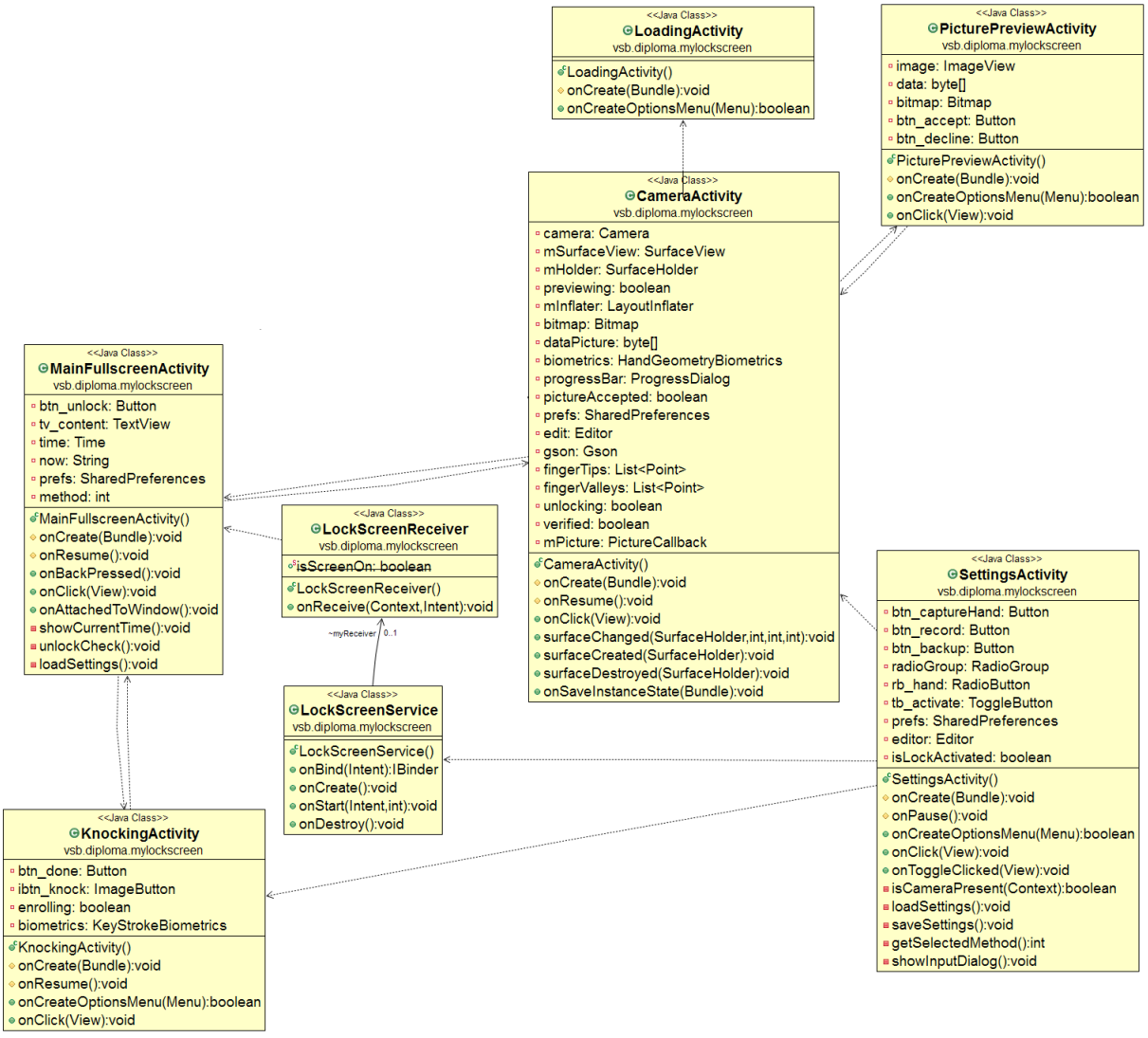
V momentě, kdy je snímání ukončeno, jsou spočítány rozdíly mezi jednotlivými časy. Tyto údaje si třída opět drží ve formě vektoru. V případě, že aplikace pracuje v registračním režimu, je tento vektor uložen jako šablona do databáze (**SharedPreferences**), k čemuž slouží metoda *SaveSample()*.

V ověřovací fázi je šablona načtena z databáze metodou *loadSample()*, spočítány rozdíly aktuálně nahraného klepání na displej a oba vzorky jsou porovnány pomocí metody *compareSamples()*. Tato metoda vrací booleovskou hodnotu (true nebo false) v závislosti na výsledku porovnávání. Metoda nejdříve jednoduše zkoumá, zda se rovnají počty klepnutí v šabloně a v aktuálním vzorku. Pokud počty nejsou stejné, výsledek je automaticky záporný. V opačném případě je vypočtena kosinová podobnost vzorkových vektorů podle rovnice 6.2 (metoda *vectorSimilarity()*), a pokud míra převyšuje stanovený práh (konstanta *REQUIRED_RATIO*), vrací metoda kladný výsledek.

Původní myšlenkou při implementaci této metody bylo využití mikrofonu zařízení pro snímání úderů uživatele. V takovém případě by uživatel nemusel klepat na displej, ale kamkoliv na zařízení. Při realizaci však docházelo k velkým problémům při zpracování zaznamenané zvukové stopy. Záměrem bylo nalézt dostatečně velké změny v úrovni hlasitosti záznamu, které by značily moment klepnutí. Problémem však bylo, že i v naprostém tichu záznam často obsahoval velké výkyvy v úrovni hlasitosti způsobené pravděpodobně kvalitou mikrofonu na zařízení používaném k testování. Z toho důvodu bylo snímání velmi nespolehlivé a nepřesné a nakonec tedy bylo od tohoto přístupu odstoupeno ve prospěch výše popsaných metod.

6.2 Struktura aplikace

Základní kameny aplikace uzamykací obrazovky tvoří 4 aktivity (**MainFullScreenActivity, SettingsActivity, KnockingActivity, CameraActivity**), jedna služba s názvem **LockScreenService**, jeden přijímač **LockScreenReceiver** a knihovna poskytující rozhraní k biometrickým metodám ověřování. Všechny aktivity pochopitelně dědí ze systémové třídy **Activity** a mohou přepisovat její metody. Stejně tak služba dědí ze systémové třídy **Service** a přijímač dědí ze systémové třídy **BroadcastReceiver** a i tyto třídy mohou přepisovat metody svých rodičů. Celou strukturu aplikace ukazuje třídní diagram, viz Obrázek 6-3.



Obrázek 6-3: Třídní diagram aplikace

6.2.1 Aktivita MainFullScreenActivity

Hlavní odpovědností aktivity s názvem **MainFullScreenActivity** je simulovat chování obrazovky reprezentující zámek displeje. Aktivita tedy zobrazuje své rozhraní uživateli podle pravidel popsaných v kapitole 5.1. Aby aktivita pracovala v celoobrazovkovém režimu, je v manifestu aplikace při definici aktivity přidán následující řádek:

```
android:theme="@android:style/Theme.NoTitleBar.Fullscreen"
```

Důležitým úkolem bylo zajistit, aby uživatel nebyl schopen aktivitu ukončit hardwarovými tlačítky. Tlačítko „Zpět“ je ošetřeno přepsáním metody *onBackPressed(),* tělo metody obsahuje pouze výraz `return`, což zajišťuje, že aktivita nereaguje na stisk tlačítka klasickým způsobem (nepřenese do popředí poslední aktivitu ze zásobníku). Ošetření reakce aktivity na stisk tlačítka „Domů“ bylo původně implementováno přepsáním metody *onKeyDown()* a detekováním tohoto tlačítka pomocí jeho kódu *(*`KeyEvent.KEYCODE_HOME`*)*. Nicméně zde bylo zjištěno, že tento kód není vždy posílán aplikaci a je ošetřen systémem, což v prostředí Android zajišťuje, že je po stisku domácího tlačítka vždy zobrazena domovská obrazovka nezávisle na aktivitě, která je aktuálně v popředí. Aby bylo aplikaci umožněno pracovat s kódem domácího tlačítka, bylo potřeba nastavit typ okna aktivity na `TYPE_KEYGUARD_DIALOG`, což zajistilo, že činnost aktivity není stiskem tlačítka „Domů“ přerušena. Bohužel toto nastavení způsobuje, že aktivita již nemůže být zobrazena přes celou obrazovku se skrytým stavovým řádkem. Android tento typ chování aplikace totiž považuje za nebezpečný (pomineme-li zámek obrazovky, je logické, že aplikace pracující v celoobrazovkovém módu bez možnosti standardního ukončení je považována za škodlivou). V tento moment se již činnost aktivity blíží skutečnému zámku obrazovky, nicméně pokud je aktivní standardní zámek obrazovky, bude aktivitu vždy „překrývat“. Pro zamezení tomuto chování jsou nastaveny další parametry okna, a to `FLAG_DISMISS_KEYGUARD` a `FLAG_SHOW_WHEN_LOCKED`.

6.2.2 Aktivita SettingsActivity

Aktivita s názvem **SettingsActivity** je uživateli zobrazena jako první při spuštění aplikace, čehož je docíleno přidáním následujícího kódu při definování aktivity v manifestu aplikace:

```
<intent-filter>
<action android:name="android.intent.action.MAIN" />
<category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
```

Aktivita poskytuje uživateli rozhraní pro základní nastavení aplikace, v prvé řadě aktivaci a deaktivaci zámku a volbu biometrické metody ověřování. Všechna uživatelská nastavení jsou uchovávána pomocí rozhraní **SharedPreferences**.

6.2.3 Aktivita KnockingActivity

Aktivita poskytuje uživatelské rozhraní pro snímání uživatelovy dynamiky klepání na dotykový displej zařízení. Drží si instanci třídy **KeyStrokeBiometrics** z biometrické knihovny a při klepnutí uživatele na vyhraněnou oblast na obrazovce volá metodu této třídy *recordBiometrics(Object data)*. Užívá také další metody uvedené třídy pro práci s biometrickými daty (např. ukládání a načítání šablony).

6.2.4 Aktivita CameraActivity

Aktivita s názvem **CameraActivity** tvoří uživatelské rozhraní pro práci s fotoaparátem zařízení, který je využíván pro zachycení snímku ruky uživatele. Aktivita si drží instanci třídy **HandGeometryBiometrics** (proměnná *biometrics*) z biometrické knihovny a využívá jejích metod ke zpracování snímku a dalším operacím spojeným s biometrickým markantem. K zobrazení náhledu fotoaparátu před zachycením snímku využívá aktivita vykreslovací povrch reprezentovaný ovládacím prvkem třídy **SurfaceView**. Přístup k tomuto ovládacímu prvku a kontrolu nad ním zajišťuje instance třídy **SurfaceHolder**, kterou získáme voláním metody *SurfaceView.getHolder()*.

Aby aktivita byla schopná správně reagovat na změny ve vykreslovaném povrchu, je potřeba, aby implementovala rozhraní **SurfaceHolder.CallBack**. Musí tedy přepisovat metodu *surfaceCreated(SurfaceHolder holder,…),* metodu *surfaceChanged(SurfaceHolder holder)* a metodu *surfaceDestroyed(SurfaceHolder holder*).

Vzhledem k tomu, že aktivita potřebuje pro svou funkčnost využívat fotoaparát zařízení, musí manifest aplikace obsahovat následující kód:

```
<uses-permission android:name="android.permission.CAMERA" />
<uses-feature
        android:name="android.hardware.camera"
        android:required="false" />
```

Díky prvnímu řádku kódu musí uživatel během instalace potvrdit, že aplikace může využívat fotoaparát zařízení. Další řádky jsou přidány, aby aplikace věděla, že potřebuje ke svému fungování fotoaparát. Instalace by tedy vůbec neproběhla na zařízení bez fotoaparátu, nicméně poslední řádek říká, že fotoaparát není nezbytný. Aplikace totiž bude schopna fungovat i na zařízeních bez fotoaparátu, její funkčnost ale bude omezena pouze na ověřování identity pomocí dynamiky úderů na displeji.

Pro přístup k fotoaparátu zařízení slouží aktivitě instance třídy *Camera*, kterou získáme voláním `Camera.open()`[15] v těle metody *surfaceCreated(...)*. Od tohoto okamžiku má přístup k danému fotoaparátu výhradně naše aplikace a žádná jiná aplikace jej nemůže využívat. Pro zahájení vykreslování náhledu fotoaparátu slouží následující kód umístěný v metodě *surfaceChanged(...)*:

```
camera.setPreviewDisplay(mHolder);
camera.startPreview();
```

První řádek nastaví vykreslovací povrch pomocí výše popisované instance třídy **SurfaceHolder** (zde proměnná s názvem *mHolder*), druhý řádek spustí samotné vykreslování do vybraného povrchu.

K zachycení snímku slouží příkaz `camera.takePicture(null, null, mPicture)`, kde *mPicture* je implementací rozhraní **PictureCallback**, přepisuje tedy povinně metodu `onPictureTaken(byte[] data, Camera camera)`, a tím umožňuje přístup k zachycenému snímku ve formě pole bytů (parametr *data*). Snímek je zde příkazem `biometrics.recordCharacteristic(data)` předán k dalšímu zpracování instanci třídy **HandGeometryBiometrics**. Poté je spuštěna aktivita s názvem **LoadingActivity**. Tím, že tato aktivita přejde do popředí, je „zničen“ vykreslovací povrch fotoaparátu, tedy je volána metoda *surfaceDestroyed(...)*.

V těle metody *surfaceDestroyed(...)* je zastaveno snímání a fotoaparát je uvolněn k užívání jiným aplikacím. Toho je docíleno následujícím kódem:

```
camera.stopPreview();
camera.release();
camera = null;
```

Aktivita detekuje pomocí proměnné *unlocking*, zda pracuje v režimu registrace nebo ověřování, a podle toho provádí v rámci metody *surfaceDestroyed(...)* další kroky. Proměnná je nastavena dle informací v objektu třídy **Intent**, kterým byla aktivita spuštěna, což zajišťuje řádek kódu:

```
unlocking = getIntent().getBooleanExtra("UNLOCKING", false);
```

Pokud aktivita pracuje v ověřovacím režimu (proměnná nastavená na hodnotu ***true***), načítá uloženou šablonu voláním `biometrics.loadSample()`. Následně porovnává šablonu s aktuálními daty voláním `biometrics.compareSamples()`. Poté je na základě výsledku porovnání spuštěna hlavní uzamykací obrazovka s dodatečnou informací o tom, zda má proběhnout odemknutí obrazovky.

---

[15] Tímto získáme přístup k prvnímu zadnímu fotoaparátu. Pokud zařízení disponuje více fotoaparáty, lze metodu volat s parametrem udávajícím ID fotoaparátu.

V případě, kdy je hodnota proměnné *unlocking* nastavena na ***false***, pracuje aktivita v registračním módu. Pomocí následujícího kódu je spuštěna aktivita **PicturePreview**, která uživateli ukáže obrázek siluety ruky s detekovanými významnými body (obrázek je vytvořen v rámci zpracování dat ve třídě **HandGeometryBiometrics** a získán pomocí volání metody v prvním řádku kódu). V rámci aktivity **PicturePreview** se uživatel může rozhodnout, zda chce takto zpracovaný snímek uložit jako šablonu.

```
Bitmap bitmap = biometrics.getBitmap();
ByteArrayOutputStream stream = new ByteArrayOutputStream();
bitmap.compress(Bitmap.CompressFormat.PNG, 100, stream);
byte [] dataPicture = stream.toByteArray();
Intent intent = new Intent(getApplicationContext(),
PicturePreview.class);
intent.putExtra("PICTURE", dataPicture);
startActivity(intent);
```

6.2.5 Služba LockScreenService

Služba s názvem **LockScreenService** je potomkem systémové třídy **Service**. Dokáže získat přístup k systémové službě, která obsluhuje standardní uzamykání obrazovky a jejím prostřednictvím tento zámek zakáže. Následující kód umístěný v přepsané metodě *onCreate()* ukazuje, jakým způsobem je toho dosaženo:

```
KeyguardManager km =
(KeyguardManager)getSystemService(KEYGUARD_SERVICE);
KeyguardManager.KeyguardLock k1 = km.newKeyguardLock("");
k1.disableKeyguard();
```

Aby služba byla schopná zakázat systémový zámek obrazovky, musí být v manifestu aplikace deklarováno oprávnění `DISABLE_KEYGUARD`.

Dalším důležitým úkolem služby je registrace přijímače **LockSreenReceiver**, jehož funkcionalita je popsána v podkapitole 6.2.6. V přepsané metodě *onDestroy()* je pak registrace přijímače zrušena. Metoda *onDestroy()* je volána v momentě, kdy je činnost služby ukončena.

6.2.6 Přijímač LockScreenReceiver

Třída s názvem **LockScreenReceiver** plní funkci přijímače a dědí ze systémové třídy **BroadcastReceiver**. Přepisuje metodu *onReceive()*, která slouží k naslouchání systémovým nebo aplikačním oznámením a umožňuje aplikaci na vybraná oznámení vhodně reagovat. **LockScreenReceiver** spouští zámek obrazovky (aktivitu **MainFullScreenActivity**) v momentě, kdy je vypnut displej zařízení (oznámení `Intent.ACTION_SCREEN_OFF`), při opětovném zapnutí displeji je tak zámek aktivní. Zámek je spuštěn také v návaznosti na zapnutí zařízení, což je

detekováno oznámením `Intent.ACTION_BOOT_COMPLETED`. V manifestu aplikace musí být deklarováno oprávnění `RECEIVE_BOOT_COMPLETED`.

LockScreenReceiver také zajišťuje, že po skončení telefonního hovoru bude zámek obrazovky opět aktivní. Android umožňuje sledovat stav telefonu v souvislosti s hovory pomocí třídy **TelephonyManager**. Třída rozlišuje tři základní stavy: Vyzvánění (`CALL_STATE_RINGING`), probíhající hovor (`CALL_STATE_OFFHOOK`) a nečinnost (`CALL_STATE_IDLE`). Pokud se změní stav telefonu, je vysíláno oznámení `TelephonyManager.ACTION_PHONE_STATE_CHANGED`. Přijímač v reakci na toto oznámení zjistí aktuální stav telefonu, a pokud je telefon ve stavu nečinnosti (je zřejmé, že předchozím stavem mohlo být jedině vyzvánění nebo probíhající hovor), spustí opět aktivitu **MainFullScreenActivity**. Pro sledování stavu telefonu musí manifest aplikace obsahovat oprávnění `READ_PHONE_STATE`.

7 Testování

Kapitola popisuje průběh testování implementovaných metod ověřování identity. Výkonnost aplikace je ohodnocena prostřednictvím vybraných chybových měr, které jsou popsány v podkapitole 2.1.3. Součástí kapitoly je představení zařízení, na němž byla aplikace testována. Nakonec je uvedeno srovnání dosažených výsledků s metodou ověřování identity uživatelů podle rozlišování obličeje, která je již v prostředí Android implementována.

7.1 Zařízení používané k testování

Po celou dobu vývoje projektu byla aplikace testována na zařízení Samsung GT-S5570, které je známé také pod názvem Samsung Galaxy mini (viz Obrázek 7-1). Na zařízení je instalován starší operační systém Android Gingerbread, konkrétně ve verzi 2.3.4. Frekvence procesoru zařízení je 600 MB, což postačuje k dostatečně rychlému zpracování všech algoritmů, které vyvíjená aplikace využívá.

Obrázek 7-1: Samsung Galaxy mini, převzato z [33]

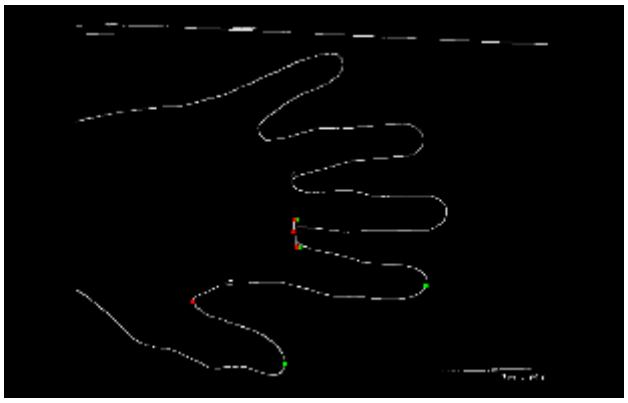
Tento chytrý telefon patří k méně výkonným zařízením, nicméně pro svou jednoduchost a nízkou cenu je rozšířen mezi podstatným množstvím uživatelů. Fotoaparát zařízení dokáže pořídit snímky s maximálním rozlišením pouhé 3 Mpix, navíc nedisponuje funkcemi blesku ani automatického ostření. Tyto nedostatky mohou mít negativní vliv především na rozpoznávání identity na základě geometrie ruky, neboť pořízené snímky nemusí mít vždy vhodnou kvalitu pro další zpracování.

7.2 Testování statické biometrické metody

Jako zástupce statických biometrických metod bylo implementováno rozpoznávání uživatele na základě geometrie jeho ruky. U této metody byla testována především schopnost aplikace správně zachytit a zpracovat snímek ruky a dále míry chybného přijetí, chybného odmítnutí, chybné shody a chybné neshody. Na konci podkapitoly jsou uvedeny souhrnné výsledky testování, viz Tabulka 7-1.

Největší překážkou při implementaci metody geometrie ruky bylo správné detekování obrysu ruky v obraze. Tento problém stále nebyl úplně vyřešen, neboť algoritmy jsou velmi citlivé na kvalitu vstupního obrazu. Zpracování je ovlivňováno způsobem, jakým je ruka na obrázku zachycena, důležitá je především povaha pozadí, velkou roli hrají také stíny na snímané ruce. Algoritmy pracují lépe, pokud je ruka zachycena proti kontrastnímu jednolitému pozadí, než v případě příliš strukturovaného nebo nekontrastního pozadí. Nicméně i v případě vhodného pozadí může zpracování obrazu selhat, což je připisováno okolnímu osvětlení a kvalitě fotoaparátu testovacího zařízení, který není schopen správně ostřit. Nejlepších výsledků při zpracování obrazu dosahuje metoda v případě, kdy je snímek ruky pořízen oproti bílé ploše monitoru (velký rozdíl jasové složky mezi rukou a pozadím), proto jsou testy prováděny za uvedených podmínek.

Prvním krokem testování statické biometrické metody je stanovení chybové míry **FTA** (míra neschopnosti nasnímat), přičemž za úspěšný výsledek snímání je považována situace, kdy aplikace správně detekuje čtyři údolí a pět špiček prstů v zachyceném snímku. Pro stanovení míry FTA bylo provedeno celkem 100 snímání, 47 z nich dopadlo neúspěšně. Míra neschopnosti nasnímat je rovna poměru všech a neúspěšných snímání, výsledkem je tedy hodnota 0.47, což značí, že systém není schopen správně nasnímat geometrii ruky ve **47 %** všech pokusů. Příklad neúspěšného nasnímání ukazuje Obrázek 7-2.



Obrázek 7-2: Špatně detekované klíčové body v obraze

Přestože algoritmus na první pohled správně extrahoval obrys ruky ze snímku, selhalo následné hledání špiček a údolí mezí prsty. Je vidět, že algoritmus správně detekoval špičku palce i ukazováčku a také údolí mezi nimi, další klíčové body však již nedokázal správně určit. Problém může být způsoben drobnými nerovnostmi a trhlinami v obryse. Za nejdelší hranu pak může být považován pouze určitý fragment obrysu ruky a detekce klíčových bodů tak nepřináší očekávaný výsledek.

Dalšími testovanými vlastnostmi implementované metody jsou míry chybného přijetí (**FAR**) a chybného odmítnutí (**FRR**). Obě míry závisí na zvoleném prahu T (konstanta `REQUIRED_RATIO` ve třídě **HandGeometryBiometrics**), který je stanoven na základě výsledků porovnávání vzorků stejného uživatele během vývoje aplikace. Pokud algoritmus dokázal správně detekovat klíčové body v obraze, bylo vypozorováno, že výsledkem porovnávání aktuálního vektoru s vektorem vzdáleností

uloženým jako šablona je hodnota přesahující 0.99. Práh je tedy stanoven na hodnotu 0.998. Takto vysoká hodnota je nastavena z toho důvodu, že geometrie ruky nevykazuje příliš velkou mezitřídní variabilitu a při porovnávání šablony se vzorky jiných uživatelů bývala často výsledkem hodnota až kolem 0.98.

Ke stanovení hodnoty míry **FAR** bylo požádáno 30 dobrovolníků, aby se pokusili odemknout obrazovku pomocí své ruky. Ani jednomu se to nepodařilo, což bylo pravděpodobně také způsobeno tím, že obraz jejich ruky nebyl správně zpracován a výsledek porovnávání tak byl automaticky záporný. Výsledná míra FAR je 0, tedy v **0 %** pokusů systém akceptoval neoprávněného uživatele.

K určení hodnoty míry **FRR** bylo provedeno celkem 50 pokusů o ověření identity uživatele, jehož šablona je uložena v databázi. Z těchto pokusů nastal případ odmítnutí celkem v 28 případech, tedy selhalo **56 %** pokusů o přijetí právoplatného uživatele. Je potřeba mít na paměti, že odmítnutí mohlo být stejně jako v případě míry FAR způsobeno neschopností správně nasnímat aktuální vzorek. Ve většině případů porovnávání bude tato skutečnost pravděpodobným důvodem odmítnutí uživatele vzhledem k vysoké míře FTA.

Pro vyjádření poměru chybných přijetí a odmítnutí neovlivněných mírou FTA, slouží míry **FMR** (míra chybné shody) a **FNMR** (míra chybné neshody). Tyto míry budou mít vysokou vypovídací hodnotu o kvalitě srovnávacího modulu aplikace, neboť berou v potaz pouze porovnávání těch vzorků, které byly korektně zpracovány. Obě míry také závisí na stanoveném prahu, stejně jako míry FAR a FRR popisované v předchozích odstavcích.

Výpočet hodnoty **FNMR** probíhal obdobným způsobem jako výpočet FRR. Byly tedy opakovaně prováděny pokusy o ověření oprávněného uživatele a bylo sledováno, zda jej systém akceptuje. Započítána byla pouze ta porovnání, při kterých byly správně detekovány klíčové body aktuálního vzorku. Z celkového počtu 50 porovnání třikrát nastala situace, kdy uživatel nebyl systémem přijat. Výsledná míra je tedy rovna hodnotě 0.06, jinými slovy v **6 %** případů je oprávněnému uživateli chybně zamítnut přístup.

Pro získání hodnoty **FMR** bylo opět požádáno 30 dobrovolníků, aby odemknuli telefon pomocí své ruky, tentokrát však bylo také sledováno, zda systém správně detekoval klíčové body v nasnímaném obraze. Pouze tyto pokusy pak byly počítány do statistiky porovnávání. Z 30 porovnání byla jako shodná chybně vyhodnocena pouze jedna dvojice porovnávaných vzorků. Výsledná hodnota FMR je tedy 0.03, což značí, že systém chybně akceptoval **3 %** neoprávněných uživatelů.

Chybová míra	Hodnota
FTA	**0.47**
FAR	**0.00**
FRR	**0.56**
FMR	**0.03**
FNMR	**0.06**

Tabulka 7-1: Výsledky testování statické biometrické metody

Z výsledků testování je zřejmé, že nejslabším místem implementované metody porovnávání podle geometrie ruky je modul extrakce klíčových znaků, který není schopen ve snímku správně rozlišit údolí a špičky prstů téměř v polovině všech pokusů. Tento jev může být do značné míry způsoben nízkou kvalitou senzorového modulu, tedy fotoaparátu zařízení. Na druhou stranu porovnávací modul vykazuje vysokou spolehlivost, neboť v drtivé většině případů systém správně přijal oprávněného uživatele a naopak odmítl pokusy neoprávněných uživatelů.

7.3 Testování dynamické biometrické metody

Z dynamických biometrických byla k implementaci vybrána metoda ověřování identity na základě dynamiky úderů na dotykovém displeji. Původně byla metoda implementována pomocí mikrofonu zařízení, ale v drtivé většině případů nebyla schopna vhodně zaznamenat zvukovou stopu tak, aby z ní mohla správně extrahovat potřebná data. Vzhledem k vysoké chybovosti tak byla nahrazena jednodušší metodou. Záznam dynamiky úderů je tak proveden pomocí tlačítka zabírajícího podstatnou část displeje, na které uživatel může při snímání libovolně klepat. V rámci testování metody budou vyjádřeny míry FTA, FAR a FRR. Souhrn výsledků uvádí Tabulka 7-2 v závěru této podkapitoly.

Současná realizace detekování úderů na displeji vykazuje při snímání bezchybné výsledky, při všech 50 pokusech o zaznamenání dynamických dat byla metoda úspěšná. Míra **FTA** je tedy u této metody rovna **0**.

Míry FAR a FRR jsou opět ovlivňovány výší stanoveného prahu, který byl na základě experimentů a pozorování v rámci vývoje aplikace nastaven na hodnotu 0.98 (konstanta `REQUIRED_RATIO` ve třídě **KeyStrokeBiometrics**). Dále jsou výsledky ovlivněny samotným vzorkem, který je uložen jako šablona. Uživatel má totiž svobodu v tom, kolikrát na displej poklepe a může tak použít libovolně dlouhý vzor. Čím delší vzor si uživatel uloží, tím větší zabezpečení mu metoda poskytne, neboť tím značně klesá pravděpodobnost prolomení. Naopak se ale zvyšuje pravděpodobnost situace, že uživatel nedokáže při odemykání správně zopakovat styl klepání, který má uložen jako šablonu.

Při určování míry **FAR** bylo 30 dobrovolníků požádáno, aby se pokusili odemknout telefon pomocí poklepání na displej. Nejdříve jim nebyly sděleny žádné informace o požadovaném počtu poklepání na displej. V tomto případě se nikomu nepodařilo zařízení odemknout. Poté byl dobrovolníkům prozrazen požadovaný počet klepnutí, ale i tak se podařilo zámek obejít pouze v jediném případě. Z těchto pokusů můžeme říci, že hodnota FAR leží v intervalu **<0.00;0.03>**.

Míra **FRR** byla stanovena na základě obdobného testování jako u statické ověřovací metody. Bylo provedeno 50 pokusů o odemknutí obrazovky právoplatným uživatelem. V případě nastavení jednoduchého vzoru klepání (pouhé 4 klepnutí) byly všechny pokusy úspěšné. V momentě, kdy byl nastaven složitější vzor (8 klepnutí na displej), bylo již složitější klepání přesně zopakovat a uživatel byl chybně zamítnut systémem v šesti případech. Hodnota míry FRR se tedy pohybuje v intervalu **<0.00;1.20>** v závislosti na složitosti uložené šablony.

Chybová míra	Hodnota
FTA	**0.00**
FAR	**0.00 – 0.03**
FRR	**0.00 – 1.20**

Tabulka 7-2: Výsledky testování dynamické biometrické metody

U metody ověřování identity podle dynamiky klepání na dotykový displej bylo obtížné stanovit přesně chybové míry, neboť výsledky jsou velmi závislé na povaze uloženého vzoru. Na rozdíl od statické metody, kde uživatelé nemají možnost příliš měnit vlastní šablonu, je u dynamické metody uživatelům poskytnuta volnost v nastavení vlastního vzoru klepání. Toto nastavení zvyšuje bezpečnost celého systému, protože pokud neoprávněný uživatel nezná přesný počet klepnutí, prakticky není schopen vzor zopakovat. I v případě, že útočník zná počet klepnutí, je malá pravděpodobnost, že dokáže napodobit styl klepání uživatele. Tato pravděpodobnost navíc ještě prudce klesá se zvyšováním složitosti uložené šablony. Samozřejmě příliš složitý vzor s sebou nese vyšší riziko chybného odmítnutí právoplatného uživatele systému.

7.4 Srovnání se zabudovaným rozpoznáváním obličeje

Android od verze 4.0 poskytuje uzamykací obrazovku využívající ověřování identity uživatele podle jeho obličeje. Zabudování této metody mělo přinést zvýšení bezpečnosti zařízení, nicméně metoda nebyla ve své první verzi příliš sofistikovaná a systém šlo jednoduše obelstít předložením fotografie. Z toho důvodu bylo v další verzi nutné mrknutí pro potvrzení živosti uživatele, ale i toto opatření šlo poměrně jednoduše obejít vytvořením z fotografie krátké videosekvence mrknutí. Bezpečnost metody nejvíce doplácí na to, že fotografie obličeje většiny z nás lze v současné době velmi lehce získat např. na sociálních sítích a tuto fotografii dále zneužít k podvedení zámku. Přesné chybové míry této metody bohužel nejsou k dispozici, a tak na jejich základě nelze provést srovnání s implementovanými metodami a musíme se omezit pouze na porovnání z hlediska bezpečnosti a použitelnosti.

Uzamykací obrazovka využívající geometrii ruky implementovaná v diplomové práci rozhodně není tak zranitelná jako rozpoznávání obličeje. Zámek sice lze také obelstít předložením fotografie ruky, nicméně pro útočníka bude mnohem složitější, v mnoha případech dokonce nemožné, takový snímek získat. V tomto ohledu poskytuje geometrie ruky lepší ochranu zařízení než obličej.

Srovnávat druhou implementovanou metodu s rozpoznáváním obličeje není úplně na místě, neboť se jedná o rozdílné typy biometrických charakteristik. Dynamika úderů na dotykovém displeji je behaviorální charakteristika a dala by se spíše přirovnat k zámku obrazovky typu *pattern*. Oproti rozpoznávání obličeje poskytuje metoda vyšší úroveň zabezpečení (hlavně u složitějších vzorů), spolehlivější snímání charakteristiky a lze předpokládat, že i nižší míru chybného odmítnutí uživatelů.

Oproti oběma implementovaným metodám má rozpoznávání obličeje vyšší nároky na samotné Android zařízení, vyžaduje totiž vyšší verzi operačního systému (4.0 a výše) a také potřebuje ke správnému fungování přední fotoaparát. Lze tedy říci, že metody implementované v rámci diplomové práce mohou najít uplatnění na podstatně širší škále zařízení.

8 Závěr

Chytrá zařízení fungující na platformě Android se již stala běžnou součástí našich životů a mnoho z nás je denně využívá i pro přístup k emailovým a dalším internetovým účtům nebo například k obsluze internetového bankovnictví. Spousta uživatelů má ve svém zařízení uložena citlivá data, a tak rostou nároky na zabezpečení zařízení. Základní ochranu může poskytnout sofistikovaný zámek obrazovky, který zamezí v užívání zařízení nepovoleným osobám. Ověřování identity pomocí biometrických charakteristik nachází čím dál tím širší využití i v běžném životě, ačkoliv klasické ověřování pomocí hesel, PIN kódů nebo přístupových karet stále převládá. Z důvodu potřeby vyššího stupně zabezpečení a pohodlí uživatelů však lze předpokládat v následujících letech masivní nástup biometrických systémů na úkor klasických metod identifikace a verifikace. Biometrie v některých případech prakticky znemožňuje krádeže identity a poskytuje uživatelům přirozenou a jednoduchou cestu k proklamování vlastní totožnosti.

V rámci diplomové práce byla vyvinuta mobilní aplikace pro Android zařízení, která může nahradit klasický zámek obrazovky a využívá biometrických metod ověřování identity. V úvodu práce byl uveden přehled nejrůznějších biometrických charakteristik a byla zhodnocena jejich použitelnost na mobilních zařízeních v prostředí Android. K implementaci byly vybrány dvě ověřovací metody, které ještě v prostředí Android nejsou implementovány, jedna jako zástupce statických ověřovacích metod, druhá jako zástupce dynamických metod ověřování. Obě metody jsou součástí samostatně vyvíjené knihovny, kterou zámek obrazovky využívá.

Jako zástupce statických metod bylo implementováno ověřování identity na základě geometrie ruky. Toto řešení je výpočetně poměrně nenáročné, a tedy dostatečně rychlé, kromě fotoaparátu nevyžaduje žádný přídavný hardware a poskytuje uživatelům pohodlný způsob snímání biometrické charakteristiky. Metoda je bezpečnější než zabudované ověřování identity podle obličeje, jelikož snímek ruky k obelstění systému nelze získat tak jednoduše jako snímek obličeje uživatele. Největším nedostatkem implementované metody je pak slabá schopnost správně detekovat klíčové body na obrysu ruky, což má často za následek nutnost provádět opakované snímání.

Dynamickou metodou implementovanou v rámci diplomové práce je ověřování identity podle dynamiky úderů na dotykovém displeji. Zabezpečení pomocí této metody je velice jednoduché, nicméně velmi spolehlivé a těžko prolomitelné. U metody nebyly zjištěny žádné nedostatky a lze předpokládat, že by se mohla stát plnohodnotnou alternativou k bezpečnostním uzamykacím obrazovkám, které jsou v současnosti v prostředí Android používány.

Největší prostor k dalšímu rozvoji aplikace tak zůstal v oblasti statické metody ověřování, jejíž algoritmy potřebují zdokonalit tak, aby byly schopny v rozumném množství scénářů správně zpracovat snímek ruky a detekovat v obraze klíčové body. V současné chvíli využívá aplikace pro správné fungování modulu detekce ruky externí knihovnu OpenCV, což má za následek nutnost instalace knihovny při instalování aplikace na koncová zařízení. V budoucnu by bylo vhodné algoritmy přepsat nativně tak, aby tato závislost byla odstraněna a bylo možno publikovat verzi

aplikace bez OpenCV. Také by bylo možné doplnit aplikaci o jiné statické metody ověřování identity, které knihovnu OpenCV nepotřebují. Implementovaná biometrická knihovna nechává možnost přidat do aplikace zámku obrazovky další metody biometrického ověřování identity prostřednictvím definovaného rozhraní.

9 Literatura

1. VACH, M. Historie biometrik a jejich využití ve výpočetní technice. *Fakulta Informatiky Masarykovy Univerzity* [online]. 2006-2014 [cit. 2014-02-03]. Dostupné z: http://www.fi.muni.cz/usr/jkucera/pv109/2003/xvach_biometriky.htm

2. JAIN, A. a KOLEKTIV. *Handbook of biometrics*. New York: Springer Science, 2008. ISBN 978-0-387-71040-2.

3. DRAHANSKÝ, M. a KOLEKTIV. *Biometrie*. Brno: Computer Press a.s. 2011. ISBN 978-80-254-8979-6.

4. JAIN, A. a KOLEKTIV. *Introduction to biometrics*. New York: Springer, 2011. ISBN 978-0-387-77325-4.

5. What is Biometrics? *Biometrics Research Group* [online]. 2009 [cit. 2014-03-20]. Dostupné z: http://biometrics.cse.msu.edu/info/index.html

6. JAIN, A. a KOLEKTIV. An introduction to biometric recognition. Michigan: IEEE, 2004, č. 1. ISSN 1051-8215.

7. Biometrics FAQ. *Bromba Biometrics* [online]. 02. 11. 2012 [cit. 2014-03-05]. Dostupné z: http://www.bromba.com/faq/biofaqe.htm

8. KARÁSEK, J. Snímač otisků prstů v Samsung Galaxy S5 bude přístupný vývojářům aplikací. *SmartMania.cz* [online]. 27. 2. 2014 [cit. 2014-03-01]. Dostupné z: http://smartmania.cz/bleskovky/snimac-otisku-prstu-v-samsung-galaxy-s5-bude-pristupny-vyvojarum-aplikaci-7086

9. Malý průvodce moderní biometrií. *ITBIZ* [online]. 16. 12. 2013 [cit. 2014-02-10]. Dostupné z: http://www.itbiz.cz/clanky/maly-pruvodce-moderni-biometrii

10. Lidské oko. *Wikipedie* [online]. 2014 [cit. 2014-03-28]. Dostupné z: http://cs.wikipedia.org/wiki/Lidsk%C3%A9_oko

11. Iris Scanners & Recognition. *findBIOMETRICS* [online]. 2014 [cit. 2014-03-30]. Dostupné z: http://findbiometrics.com/solutions/iris-scanners-recognition/

12. Gait Recognition. *Global Security Intelligence* [online]. 2014 [cit. 2014-02-10]. Dostupné z: http://globalseci.com/?page_id=44

13. ARUNA. How to Implement the Voice Recognition Functionality in Android Devices? *Contus - Technology at IT's Best* [online]. 5. 4. 2013 [cit. 2014-03-02]. Dostupné z: http://blog.contus.com/

how-to-implement-the-voice-recognition-functionality-in-android-devices/

14. DNA. *Wikipedie* [online]. 2014 [cit. 2014-02-18]. Dostupné z: http://cs.wikipedia.org/wiki/DNA

15. Android, the world's most popular mobile platform. *Android Developers* [online]. 2012 [cit. 2014-02-15]. Dostupné z: http://developer.android.com/about/index.html

16. Alliance Members. *Open Handset Alliance* [online]. 2013 [cit. 2014-02-18]. Dostupné z: http://www.openhandsetalliance.com/oha_members.html

17. VÁVRŮ, J. a M. UJBÁNYAI. *Programujeme pro Android*. Praha: Grada Publishing, a.s. 2013. ISBN 978-80-247-4863-4.

18. Android Architecture. *eLinux.org* [online]. 13. 6. 2011 [cit. 2014-02-21]. Dostupné z: http://elinux.org/Android_Architecture

19. Application Fundamentals. *Android Developers* [online]. 2012 [cit. 2014-03-15]. Dostupné z: http://developer.android.com/guide/components/fundamentals.html

20. Mobile Device Security. *OIT Help - University of Notre Dame* [online]. 2014 [cit. 2014-02-19]. Dostupné z: http://oithelp.nd.edu/information-security/stay-secure/mobile-device-security/

21. Introducing Ice Cream Sandwich. *Android* [online]. 2011 [cit. 2014-04-12]. Dostupné z: http://www.android.com/about/ice-cream-sandwich/

22. AHMAD, S. S. 5 Best Fingerprint Lock Screen FREE Android Apps. *geekOmad - Technology Blog* [online]. 8. 5. 2013 [cit. 2014-03-05]. Dostupné z: http://www.geekomad.com/2013/08/5-best-fingerprint-lock-screen-free-apps.html

23. DUBEY, K. 5 Best Fingerprint Lock Apps for Android. *TechShout* [online]. 7. 6. 2013 [cit. 2014-03-04]. Dostupné z: http://www.techshout.com/features/2012/02/best-fingerprint-lock-apps-for-android/

24. ANITA, G. Android Phones Expected to Feature Biometrics, Fingerprint Sensors Within Six Months. *Paste* [online]. 3. 10. 2013 [cit. 2014-03-02]. Dostupné z: http://www.pastemagazine.com/articles/2013/10/android-phones-expected-to-feature-biometrics-fing.html

25. VRANKULJ, A. BIO-key and InterDigital to launch OpenID-based multifactor authentication for Android. *BiometricUpdate.com* [online]. 19. 2. 2014 [cit. 2014-03-01]. Dostupné z: http://www.biometricupdate.com/201402/bio-key-and-interdigital-to-launch-openid-based-multifactor-authentication-for-android

26. Biometric Security software for iOS, Android and more. *Mobbeel* [online]. 2014 [cit. 2014-02-10]. Dostupné z: http://www.mobbeel.com/

27. Secure BRaVe App. *Fulcrum Biometrics* [online]. 2012 [cit. 2014-02-16]. Dostupné z: http://www.fulcrumbiometrics.com/BRaVe-App-p/105100.htm#

28. BSD licence. *Wikipedie* [online]. 2012 [cit. 2014-02-13]. Dostupné z: http://cs.wikipedia.org/wiki/BSD_licence

29. About. *OpenCV* [online]. 2013 [cit. 2014-02-09]. Dostupné z: http://opencv.org/about.html

30. Biometric Education. *Rosistem Barcode* [online]. 2003 - 2014 [cit. 2014-04-10]. Dostupné z: http://www.barcode.ro/tutorials/biometrics/hand_geometry.html

31. Biometric Hand Geometry Reader HandKey II. *APIS - biometric, OCR and RFID identification systems* [online]. 2011 - 2013 [cit. 2014-04-20]. Dostupné z: http://www.biometria.sk/en/hk2.html

32. WONG, K. D. Canny Edge Detection Auto Thresholding. *Kerry D. Wong* [online]. 7. 5. 2009 [cit. 2014-02-27]. Dostupné z: http://www.kerrywong.com/2009/05/07/canny-edge-detection-auto-thresholding/

33. Galaxy mini - PŘEHLED. *SAMSUNG* [online]. 1995 - 2014 [cit. 2014-04-15]. Dostupné z: http://www.samsung.com/cz/consumer/mobile-phone/mobile-phone/touchphone/GT-S5570AAAXEZ

34. 2252/2004, nařízení Rady ES o normách pro bezpečnostní a biometrické prvky v cestovních pasech a cestovních …. Dostupné také z: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CONSLEG:2004R2252:20090626:CS:PDF]

35. forenzní, forensní. *ABC.cz: slovník cizích slov* [online]. 2005-2014 [cit. 2014-02-11]. Dostupné z: http://slovnik-cizich-slov.abz.cz/web.php/slovo/forenzni-forensni

36. About The FIDO Alliance. *FIDO Alliance* [online]. 2014 [cit. 2014-02-14]. Dostupné z: https://fidoalliance.org/about

37. FIDO Alliance. *Wikipedie* [online]. 2014 [cit. 2014-03-03]. Dostupné z: http://en.wikipedia.org/wiki/FIDO_Alliance

38. CLARY, R. E-passports spread to half the globe. *SecureIDNews* [online]. 28. 2. 2012 [cit. 2014-02-10]. Dostupné z: http://secureidnews.com/news-item/e-passports-spread-to-half-the-globe

39. COURTNEY, C. Use of Biometric Security Technology at Airports Raises Concerns. *Newsmax.com* [online]. 31. 12. 2013 [cit. 2014-02-06]. Dostupné z: http://www.newsmax.com/

Newsfront/airports-biometrics-technology-automated/2013/12/31/id/544603

40. CUTLACK, G. What is Google Play? *TechRadar* [online]. 23. 3. 2012 [cit. 2014-03-01]. Dostupné z: http://www.techradar.com/news/phone-and-communications/mobile-phones/what-is-google-play-1073348

A. Seznam elektronických příloh

K diplomové práci je přiložen kompaktní disk, který obsahuje všechny přílohy této práce.

Adresářová struktura disku je následující:

- App
    - BiometricsLibrary – adresář obsahující zdrojové kódy biometrické knihovny.
    - MyLockScreen – adresář obsahující zdrojové kódy aplikace zámku obrazovky.
- Docs – adresář obsahující uživatelskou příručku aplikace a samotný text práce.